Downloaded from orbit.dtu.dk on: May 04, 2023

DTU Library

Uncertainty Quantification with Applications to Engineering Problems

Bigoni, Daniele

Publication date:
2015

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):
Bigoni, D. (2015). *Uncertainty Quantification with Applications to Engineering Problems*. Technical University of Denmark. DTU Compute PHD-2014 No. 359

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Uncertainty Quantification with Applications to Engineering Problems

Daniele Bigoni

Kongens Lyngby 2014
PHD-2014-359

Technical University of Denmark
Applied Mathematics and Computer Science
Building 324, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253031, Fax +45 45253031
compute@compute.dtu.dk
www.compute.dtu.dk

PHD: ISSN 0909-3192

Summary

The systematic quantification of the uncertainties affecting dynamical systems and the characterization of the uncertainty of their outcomes is critical for engineering design and analysis, where risks must be reduced as much as possible. Uncertainties stem naturally from our limitations in measurements, predictions and manufacturing, and we can say that any dynamical system used in engineering is subject to some of these uncertainties.

The first part of this work presents an overview of the mathematical framework used in Uncertainty Quantification (UQ) analysis and introduces the spectral tensor-train (STT) decomposition, a novel high-order method for the effective propagation of uncertainties which aims at providing an exponential convergence rate while tackling the curse of dimensionality. The curse of dimensionality is a problem that afflicts many methods based on meta-models, for which the computational cost increases exponentially with the number of inputs of the approximated function – which we will call dimension in the following.

The STT-decomposition is based on the Polynomial Chaos (PC) approximation and the low-rank decomposition of the function describing the Quantity of Interest of the considered problem. The low-rank decomposition is obtained through the discrete tensor-train decomposition, which is constructed using an optimization algorithm for the selection of the relevant points on which the function needs to be evaluated. The selection of these points is informed by the approximated function and thus it is able to adapt to its features. The number of function evaluations needed for the construction grows only linearly with the dimension and quadratically with the rank.

In this work we will present and use the functional counterpart of this low-rank decomposition and, after proving some auxiliary properties, we will apply PC

on it, obtaining the STT-decomposition. This will allow the decoupling of each dimension, leading to a much cheaper construction of the PC surrogate. In the associated paper, the capabilities of the STT-decomposition are checked on commonly used test functions and on an elliptic problem with random inputs.

This work will also present three active research directions aimed at improving the efficiency of the STT-decomposition. In this context, we propose three new strategies for solving the ordering problem suffered by the tensor-train decomposition, for computing better estimates with respect to the norms usually employed in UQ and for the anisotropic adaptivity of the method.

The second part of this work presents engineering applications of the UQ framework. Both the applications are characterized by functions whose evaluation is computationally expensive and thus the UQ analysis of the associated systems will benefit greatly from the application of methods which require few function evaluations.

We first consider the propagation of the uncertainty and the sensitivity analysis of the non-linear dynamics of railway vehicles with suspension components whose characteristics are uncertain. These analysis are carried out using mostly PC methods, and resorting to random sampling methods for comparison and when strictly necessary.

The second application of the UQ framework is on the propagation of the uncertainties entering a fully non-linear and dispersive model of water waves. This computationally challenging task is tackled with the adoption of state-of-the-art software for its numerical solution and of efficient PC methods. The aim of this study is the construction of stochastic benchmarks where to test UQ methodologies before being applied to full-scale problems, where efficient methods are necessary with today's computational resources.

The outcome of this work was also the creation of several freely available Python modules for Uncertainty Quantification, which are listed and described in the appendix.

Resumé

En systematisk kvanticering af usikkerheder der påvirker dynamiske systemer og den følgende karakterisering af usikkerhed i resultater er kritisk for tekniske designs og analyser, hvor risiko skal reduceres mest muligt. Usikkerheder kan stamme fra begrænsninger i målemetoder, tilnærmelser og fremstillingsprocesser, og alle dynamiske systemer der anvendes til ingeniørmæssige formål er under påvirkning af sådanne usikkerheder.

I den første del af dette arbejde gives en oversigt over den matematiske baggrund for anvendelsen af kvantificering af usikkerhed (På engelsk: "Uncertainty Quantication" (UQ)) til analyse formål. Der gives en introduktion til metoden "Spectral Tensor-Train (STT) decomposition" - en ny højere-ordens metode til effektive beregninger hvor man ønsker at tage højde for udbredelse af usikkerheder. Metoden søger at opnå en eksponentiel konvergenshastighed, samtidig med at den reducerer virkningen af dimensionernes forbandelse (på engelsk: "curse of dimensionality"). Dimensionernes forbandelse er et problem der opstår ved brug af mange metoder, som er baseret på meta-modeller, for hvilke beregningstiden vokser eksponentielt med antallet af input variable ved approksimation af funktioner af mange variable - et tal som vi i det følgende benævner "dimension".

Metoden bag STT dekomposition er baseret på "Polynomial Chaos" (PC) approximation og en udvikling af en lavere ordens dekomposition af funktionen til beskrivelse af den størrelse (På engelsk: "Quantity of Interest") man er interesseret i at bestemme for et givet problem. Bestemmelsen af en lavers ordens dekomposition for en funktion af mange variable bestemmes ved en diskret tensor-train decomposition, der konstrueres ved brug af en optimeringsalgoritme. Optimeringsalgoritmen sørger for at vælge punkter hensigtmæssigt ud fra egenskaberne i den tilnærmede funktion. Det nødvendige antal funktions-

beregninger vokser kun lineært med dimensionen og kvadratisk med udviklingens orden.

I dette arbejde præsenterer og anvender vi et funktionelt modstykke til udviklingen af en sådan lavere ordens dekomposition, og efter at have bevist nogle hjælpe-egenskaber, vil vi anvende PC på den, hvorved vi finder STT dekompositionen. Dette muliggør en adskillelse af de enkelte dimensioner, og leder til en meget billigere konstruktion af PC surrogat modellen. I min reference bliver egnetheden af STT dekompositionen afprøvet på nogle almindeligt anvendte test funktioner og på et elliptisk problem med stokastisk input.

I dette arbejde præsenteres endvidere tre aktive forskningsretninger som tjener til at forbedre effektiviteten af STT dekompositionen. I forbindelse hermed foreslår vi tre nye strategier til løsningen af "the ordering problem" som har til formål at forbedre effektiviteten af en tensor-train dekomposition med henblik på beregne fejlestimater baseret på sædvanligt anvendte normer der anvendes ved UQ analyse og til brug for at muliggøre anisotropisk tilpasningsning i metoden.

I anden del af arbejdet præsenteres tekniske ingeniørmæssige anvendelser af UQ metoder. I begge anvendelser indgår funktioner, hvis beregninger er tidskrævende når de evalueres og som følge heraf vil UQ analyse nyde stor fordel af anvendelse af beregningsmetoder der kun kræver få funktionsevalueringer.

Vi undersøger først usikkerheder og følsomheder i tre jernbanedynamiske problemer, hvor komponenters egenskaber i affjedringen er behæftet med usikkerheder. I disse beregninger anvendes oftest PC metoder, og gør brug af "random sampling" metoder til sammenligning af resultater eller når det er strengt nødvendigt.

I den anden anvendelse af UQ metoden undersøges udviklingen af usikkerheder, som indgår i en kompleks ikke-linear og dispersiv vandbølgemodel. Denne krævende beregning behandles med en anvendelse af state-of-the-art software til den numeriske løsning og af effektive PC metoder. Målet for denne undersøgelse er at udvikle nye stokastiske tests (på engelsk: "benchmarks") der kan bruges til at teste UQ metoder, før de anvendes på fuld-skala problemer, hvor effektive metoder er nødvendige med de beregningsressourcer der er til rådighed i dag.

I dette arbejde indgår endvidere udvikling af adskillige frit tilgængelige Python moduler til brug ved Uncertainty Quantification. De er opstillet og beskrevet i et appendiks.

Preface

This thesis was prepared at the Technical University of Denmark in fulfillment of the requirements for acquiring the PhD degree. The work has been carried out in the period between December 2011 and December 2014 at the Scientific Computing section of the Department of Applied Mathematics and Computer Science.

The project has been supervised by Associate Professor Allan Peter Engsig-Karup and Professor Jan Hesthaven (École Polytechnique Fédérale de Lausanne). The project has benefited from the external supervision by Emeritus Associate Professor Hans True.

Part of the work has been carried out during my research visit from June to December 2013 to the Uncertainty Quantification group in the Department of Aeronautics and Astronautics of the Massachusetts Institute of Technology, Boston, USA. The visit was hosted by Associate Professor Youssef M. Marzouk and partially sponsored by the Idella Foundation for which I am grateful.

The scholarship for this project was funded by the department of Applied Mathematics and Computer Science of the Technical University of Denmark.

Lyngby, December 2014

Daniele Bigoni

Publications included in the thesis

[1] D. Bigoni, A. P. Engsig-Karup, and H. True. "Comparison of Classical and Modern Uncertainty Quantification Methods for the Calculation of Critical Speeds in Railway Vehicle Dynamics". In: *13th mini Conference on Vehicle System Dynamics, Identification and Anomalies*. Budapest, Hungary, 2012

[2] D. Bigoni, A. P. Engsig-Karup, and H. True. "Anwendung der Uncertainty Quantification bei eisenbahndynamischen problemen". In: *Z E Vrail - Glasers Annalen* 137.SPL.ISSUE (2013), pp. 152–158. ISSN: 1618-8330

[3] D. Bigoni, A. P. Engsig-Karup, and H. True. "Modern Uncertainty Quantification Methods in Railroad Vehicle Dynamics". In: *ASME 2013 Rail Transportation Division Fall Technical Conference*. Altoona: ASME, Oct. 2013, V001T01A009. ISBN: 978-0-7918-5611-6. DOI: 10.1115/RTDF2013-4713

[4] D. Bigoni, H. True, and A. P. Engsig-Karup. "Sensitivity Analysis of the critical speed in railway vehicle dynamics". In: *23rd IAVSD Symposium on Dynamics of Vehicles on Roads and Tracks*. step C. Qingdao, 2013

[5] D. Bigoni, H. True, and A. P. Engsig-Karup. "Sensitivity analysis of the critical speed in railway vehicle dynamics". In: *Vehicle System Dynamics* May 2014 (Apr. 2014), pp. 272–286. ISSN: 0042-3114. DOI: 10.1080/00423114.2014.898776

[6] D. Bigoni, A. P. Engisg-Karup, and H. True. "Global Sensitivity Analysis of Railway Vehicle Dynamics on Curved Tracks". In: *Volume 2: Dynamics, Vibration and Control; Energy; Fluids Engineering; Micro*

and Nano Manufacturing. Copenhagen, Denmark: ASME, July 2014, V002T07A023. ISBN: 978-0-7918-4584-4. DOI: 10.1115/ESDA2014-20529

[7] H. True, A. Engsig-Karup, and D. Bigoni. "On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics". In: *Mathematics and Computers in Simulation* 95 (Jan. 2014), pp. 78–97. ISSN: 03784754. DOI: 10.1016/j.matcom.2012.09.016

[8] D. Bigoni, A. P. Engsig-Karup, and C. Eskilsson. "A Stochastic Nonlinear Water Wave Model for Efficient Uncertainty Quantification". In: *Journal of Engineering Mathematics (Submitted)* (Oct. 2014), p. 26. arXiv: 1410.6338

[9] D. Bigoni, A. P. Engsig-Karup, and Y. M. Marzouk. "Spectral tensor-train decomposition". In: *(Submitted)* (2014), p. 28. arXiv: 1405.5713

[10] D. Bigoni. *Spectral Toolbox*. https://launchpad.net/spectraltoolbox. 2014

[11] D. Bigoni. *Tensor Toolbox*. https://launchpad.net/tensortoolbox. 2014

[12] D. Bigoni. *UQ Toolbox*. https://launchpad.net/uqtoolbox. 2014

Acknowledgements

This work would not have been possible without the contribution of a number of people which I will try to thank here.

First I would like to express my gratitude to Associate Professor Allan Peter Engsig-Karup for proposing me such an interesting project and for his invaluable supervision. While continuously providing constructive suggestions, his guidance has always left me with much space for exploring new directions.

I would like to thank Professor Jan S. Hesthaven for his willingness in providing very constructive suggestions. I had the pleasure to meet and talk with Prof. Hesthaven in several occasions during the project, and in none of these occasions I left the meeting without a new idea to work on.

Next I would like to thank Associate Professor Hans True for the never ending enthusiasm that he puts in research and for his precious guidance along this project. His guidance often reached beyond the mere scientific aspects, but spanned also to cultural aspects which are very important when living in a foreign country.

I would like to express my gratitude to Associate Professor Youssef M. Marzouk for allowing me to spend six months in one of the most inspiring place I have ever been to. The UQgroup under his guidance is a thrilling environment where the high preparation level of the group does not preclude the discussion of the most basic concepts. I was able to share my ignorance and from this I learned a lot.

I am very grateful for the funding that I received along the years from DTU Compute, the Otto Mønsted Fond and the Idella Foundation.

I would like to thank all the colleagues who shared with me the burden of the rewarding but tiring work of doing a PhD.

I thank my friends, who I saw far less than I would have, but who I carry wherever I go. It is amazing to be in your company.

I thank my parents Roberta and Aldino and my sister Valentina for giving a meaning to the word “home”.

And I thank Stefania, *ch’al cor gentil ratto s’apprende*[1].

Thank you,

Daniele

[1]Dante Alighieri, *Divina Commedia, Inferno, Canto V.* (XIV century)

Contents

List of Symbols

$(\cdot,\cdot)_{L^2_\pi(S)}$ Inner product w.r.t. π, page 133

$(\cdot,\cdot)_{L^2(\Omega,\mathscr{F},P)}$ Inner product of random variables w.r.t. P., page 133

2^Ω Power set of Ω, page 129

$\bar{\mu}_X$ Sample mean of X, page 36

ℓ Log-likelihood, page 28

$\mathbf{E}\left[f\right]_{\pi_\mathbf{x}}$ Expectation of f under the $\pi_\mathbf{x}$ measure, page 135

$\mathbf{E}\left[X \in A|Y\right]$ Expectation of X given Y, page 137

$\Gamma(k,\theta)$ Gamma distribution, page 131

λ Lebesgue measure, page 130

$\mathbb{P}_\mathbf{N}$ Polynomials of degree up to $\mathbf{N}$, page 41

$\mathbf{x}$ Spatial variable, page 127

$\mathcal{B}$ Boundary differential operator, page 127

$\mathcal{B}e(\alpha,\beta)$ Beta distribution, page 131

$\mathcal{C}^k(S)$ Class of k differentiable functions, page 134

$\mathcal{H}^k_\pi(S)$ k-th Sobolev space, page 134

$\mathcal{L}$ Differential operator, page 127

$\mathcal{N}(\mu,\sigma^2)$ Normal distribution, page 131

$\mathcal{P}_N$ Projection operator of order N, page 39

$\mathcal{Q}_N$ One dimensional Gauss-type quadrature rule, page 40

$\mathcal{Q}_N$ Quadrature rule of order N, page 140

$\mathcal{U} = \mathcal{B}e(1,1)$ Uniform distribution, page 131

$\mathscr{B}$ Borel σ-algebra, page 130

$\mathscr{F}$ σ-algebra, page 130

μ_X Expectation of X, page 135

Ω Space of events, page 129

∂D Boundaries of spatial domain, page 127

$\pi[X \in A|Y]$ Probability of X given Y, page 136

π Measure, page 130

π_f Shorthand for $\pi_{f\circ\mathbf{x}}$, page 31

$\rho_X(x)$ Probability Density Function of X, page 132

$\rho_{X|Y}(x)$ Probability Density Function of X given Y, page 137

σ_X^2 Variance of X, page 135

$\sim$ $X \sim \pi$ means that X has probability distribution π, page 131

$\|\cdot\|^2_{\mathcal{H}^k_\pi(S)}$ k-th Sobolev norm, page 134

$\|\cdot\|_F$ Frobenious norm, page 58

$\|\cdot\|_{L^p(\Omega,\mathscr{F},P)}$ L^p norm for random variables, page 133

$|\cdot|_{S,\pi,k}$ k-th Sobolev semi-norm, page 134

$\|f\|_{L^p_\pi(S)}$ L^p norm w.r.t. π, page 133

$|\mathbf{i}|$ $|\mathbf{i}| = i_1 + \ldots + i_{d_\mathrm{s}}$, page 39

$|\mathbf{j}|_0$ $|\mathbf{j}|_0 = \max(\mathbf{j})$, page 40

$\{\cdot\}_{\mathbf{i}=\mathbf{k}}^{\mathbf{N}}$ for all $\mathbf{i} \in \{\mathbf{i} : k_j \leq i_j \leq N_j,\ \forall j \in [1,\ldots,d_\mathrm{s}]\}$, page 37

D Spatial domain, page 127

$D^{(\mathbf{i})}f$ $\mathbf{i}$-th weak derivative of f, page 134

d_s Dimension of the parameter space, page 21

$f^{(i)}$ **i**-th strong derivative of f, page 134

$F_X(x)$ CDF of X, page 131

$F_{X|Y}$ Cumulative Distribution Function of X given Y, page 137

$L(\mathbf{X})$ Likelihood of parameters $\mathbf{X}$, page 92

$L^p(\Omega, \mathscr{F}, P)$ L^p space of $\mathbb{R}$-valued random variables, page 132

$L^p(\Omega, \mathscr{F}, P; \mathbb{X})$ L^p space of $\mathbb{X}$-valued random variables, page 132

$L^p_\pi(\mathbb{R})$ L^p space of functions, page 133

N_{KL} Truncation of Karhunen-Loève expansion, page 23

$P[A|\mathscr{G}]$ Probability of A given $\mathscr{G}$, page 136

P Probability measure, page 130

S Parameter space, page 21

t Time variable, page 125

corr Correlation, page 135

cov Covariance, page 135

$\mathbf{C_X}$ Covariance of $\mathbf{X}$, page 135

E Expectation, page 135

i Multi-index $(i_1, \ldots, ß_{d_s})$, page 37

V Variance, page 135

$\mathbf{X}(\omega)$ Realization of $\mathbf{X}$, page 132

$\mathbf{X}^{(j)}(\omega)$ j-th sample of $\mathbf{X}$, page 132

$\mathrm{TS}(i)$ Total Sensitivity index of input X_i, page 87

a.e. Almost everywhere, page 130

ANOVA ANalysis Of VAriance, page 82

BVP Boundary Value Problem, page 127

CDF Cumulative Distribution Function, page 131

d.o.f. Degrees of freedom, page 18

DE Differential Equation, page 125

DO Dynamically Orthogonal, page 56

DYTSI DYnamics Train SImulation, page 101

FTT Functional tensor-train, page 58

gPC Generalized Polynomial Chaos, page 42

HDMR High Dimensional Model Representation, page 80

i.i.d. Independent and identically distributed, page 132

IVP Initial Value Problem, page 126

KDE Kernel Density Estimation, page 29

KL Karhunen-Loève, page 22

LHC Latin Hyper Cube, page 37

MC Monte Carlo, page 35

MCMC Markov Chain Monte Carlo, page 93

MEgPC Multi-Element generalized Polynomial Chaos, page 49

MWR Mean Weighted Residual, page 41

ODE Ordinary Differential Equation, page 125

PC Polynomial chaos, page 38

PCA Principal Components Analysis, page 26

PDE Partial Differential Equation, page 126

PDF Probability Density Function, page 132

PGD Proper Generalized Decomposition, page 56

POD Proper Orthogonal Decomposition, page 56

QMC Quasi-Monte Carlo, page 38

QoI Quantity of Interest, page 20

QTT Quantics tensor-train, page 63

RNG Pseudo-random Number Generator, page 33

STT Spectral tensor-train, page 56

SWE Shallow Water Equations, page 112

TSI Total Sensitivity Indices, page 87

TSP Traveling Salesman Problem, page 72

TT Tensor-train, page 57

UQ Uncertainty Quantification, page 6

To them, I said, the truth would be literally nothing but the shadows of the images.

The Republic
Plato, 380 B.C.

It is Mâyâ, the veil of deception, which blinds the eyes of mortals, and makes them behold a world of which they cannot say either that it is or that it is not.

The World as Will and Representation
A. Schopenhauer, 1818

The probability wave [..] introduces something standing in the middle between the idea of an event and the actual event, a strange kind of physical reality just in the middle between possibility and reality.

W. Heisenberg

I cannot refute your opinion that quantum theory is a complete theory of phenomena [..] but I do not share your faith that quantum theory is a complete theory of reality.

A. Einstein

Chapter 1

Introduction

This work deals with the quantification of uncertainties in engineering analysis obtained through numerical simulations. The focus is on the theoretical framework and on its application to engineering problems.

In Part I the mathematical development of UQ will be presented. UQ is a quickly growing field and a very active research area. This work aims at giving a general introduction while delving deeper on the topics used for practical applications. The mathematical theory will be presented with a broad perspective, referring the interested reader to significant literature. Chapter 2 will present the general framework of uncertainty quantification, explaining in broad terms what the goals of each of its components are. Chapter 3 introduces dynamical systems with random inputs. Chapter 4 will cover the delicate topic of the characterization of the sources of uncertainty. Since the material presented will not be used in applications, this chapter will only cover some basic techniques and include references to more advanced works. Chapter 5 will present methods for the forward propagation of uncertainty, covering both old workhorses, recent developments and the novel Spectral Tensor-Train decomposition [Bigoni et al., 9] described in section 5.2.4.2. These methods will be used on several applications in the following chapters. Chapter 6 will present global sensitivity analysis, based on the Sobol's indices, which will be applied to a railway vehicle dynamics problem. For completeness of exposition, chapter 7 covers broadly the topic of probabilistic inverse problems.

Much of the notation used in Part I conforms with the notation commonly used in literature. Nevertheless, appendices A and B provide an introduction to the topics of dynamical systems, probability theory and functional spaces, with the the necessary notational definitions.

Part II will present engineering applications of UQ. Chapter 9 will present applications of novel techniques for the forward propagation of uncertainty and sensitivity analysis on the dynamics of railway vehicles with uncertain suspension coefficients. Chapter 10 will focus on the application of methods for the forward propagation of uncertainties on a computationally intensive water wave model.

Part III contains appendices to the topics covered in this work.

The software developed along the project has been collected in three main Python modules: the `SpectralToolbox`[1], the `UQToolbox`[2] and the `TensorToolbox`[3], plus a number of smaller modules listed on the author's personal web-page[4]. Chapter D gives a short overview of these software packages along with some examples.

1.1 Why quantifying uncertainties?

Uncertainties have been troubling the humanity since its appearance on the earth. Philosophers first talked about uncertainties in the debate over reality and senses. The first thinkers setting the problem into words were the Greek geometers, who were stunned by the human ability of abstraction and of thinking about perfect forms. Plato (428-347 B.C.) sustained that abstract forms were the reality itself, and humans were only exposed to it through sensations which blurred the real objects like shadows in a cave [13]. Aristotle (384-322 B.C.) framed the subject into a more formal line of reasoning: he developed the Hylomorphism theory [14], where he separated matter and form, and gave to the latter a fundamental "holy" role, proving its priority over the matter. The discussion was carried on over the centuries and revisited for example by Arthur Schopenhauer (1788-1860) who argued the existence of the "Mâyâ veil" which misleads our senses [15]. The problem gained popularity during the last century thanks to "the Copenhagen interpretation of Quantum Theory" sustained, among others, by Werner Heisenberg, Niels Bohr and Max Born. This interpretation conjectured the probabilistic nature of reality, a point of view which notoriously unsettled Albert Einstein. Einstein spent a good part of his

[1] https://pypi.python.org/pypi/SpectralToolbox/
[2] https://pypi.python.org/pypi/UQToolbox/
[3] https://pypi.python.org/pypi/TensorToolbox/
[4] http://www2.compute.dtu.dk/~dabi/

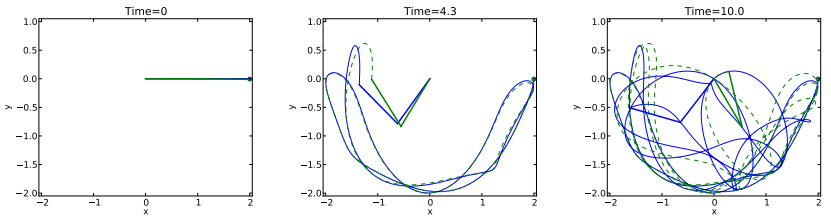
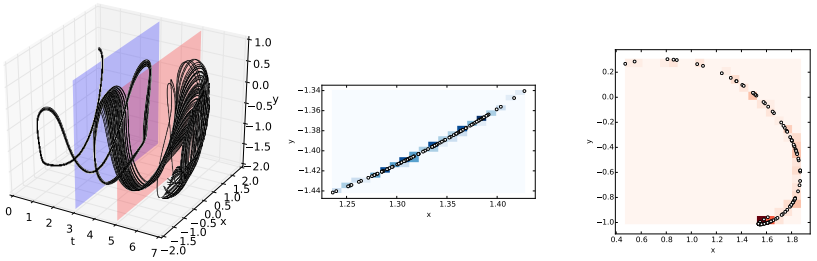


Figure 1.1: The double pendulum is one of the simplest examples of chaos: two rods are attached in series and they are let swinging. The figures show the path of the double pendulum started from two very close positions. We can see that despite the starting positions are very close, the obtained trajectories differ significantly after just 10s.

life devising paradoxes aiming at proving the incompleteness of the theory. The debate between Einstein and Bohr that originated over these paradoxes gave way to the spring of a number of different interpretations of how Quantum mechanics informs our understanding of the nature. The matter is not settled yet.

In this work we will not add a voice into the debate on reality, but we will take a Platonic view of the world: we are unable to observe and/or predict nature accurately. In practice this is due to several causes: the limited accuracy of measurement instruments, the exceedingly high cost of performing accurate experiments and/or measurements, the lack of accurate models or the computational need of using simplified ones. In the field of engineering, predictions are the result of a combination of measurements, modeling and simulations. Analysis are regularly performed on the basis of these predictions and the standard assumption made is that reality will not move significantly away from them. Using the terminology from non-linear dynamics, this means that small perturbations of the system will cause only small perturbations of the predictions. We can find a number of examples in nature where this is not the case. The most disruptive example is chaos, where small perturbations of the system lead to completely different dynamics – see figure 1.1 for an example. A notorious example of a chaotic system is the atmospheric weather, for which Lorenz [16] applied a simplification of the Saltzman's model [17], characterized by the presence of strange attractors and coined the now largely misused expression "butterfly effect".

In general non-linear systems show complex behaviors when perturbed and this can lead to unforeseen effects, with the consequence of compromising the particular outcome of an engineering analysis. In engineering, the objectives of such analysis are often related to the minimization of costs, the maximization of revenue, the improvement of safety factors, etc., and unforeseen behaviors



(a) Trajectories (b) $t = 3$ (c) $t = 5$

Figure 1.2: Continuation of the double pendulum example shown in figure 1.1. Figure 1.2a shows 1000 trajectories of the system with perturbed initial conditions. Figures 1.2b and 1.2c show the distribution of such trajectories at different times.

can have a dramatic impact on them.

In the context of this work, a perturbation is not to be merely intended as an external impulse to the system, but rather as a lack of knowledge on some property of the system. The case presented in figure 1.1 shows an example about the lack of knowledge regarding the exact initial position of the two rods. Analogously we could have considered the lack of knowledge of the exact lengths and masses of the two rods. This interpretation of perturbed systems is the reason behind the adoption of probability theory for the description of the possible outcomes of engineering analysis.

Uncertainty Quantification (UQ) is the mathematical field devoted to the description of uncertainties in dynamical systems. In the context of this work uncertainties are to be intended as lack of knowledge. In Part I we will be talking about several kinds of uncertainties. In the example in figure 1.1 the uncertainty on the initial position of the rods belongs to the input uncertainties, while the uncertainty on the trajectory of the pendulum belongs to the output uncertainties. Probabilities will help us describing these uncertainties, thus enhancing the capabilities of engineering analysis, enabling the expression of the likelihood of an event to happen. Figure 1.2 gives a glimpse to the forward propagation of uncertainties: the initial position is described by a particular probability distribution from which samples are drawn. Different trajectories are computed and the distribution of them are obtained. We can see that the strong sensitivity of the double pendulum to the initial conditions determines a rapid spread of the initially concentrated distribution.

Considering uncertainties on a system, we have actually ignored a non-negligible assumption: the mathematical model used to describe the natural system is correct. In spite of having physical evidences about the outcome of systems,

many of the commonly adopted models are based on certain levels of approximation, introduced to make them better manageable. Moreover, as mentioned in the previous discussion about quantum mechanics, we seem to be unable to provide a deterministic model even for the most detailed of the systems. Even if the model uncertainty is a critical problem for many engineering applications, in this work we will not address it directly. In the description of UQ we will talk about the construction of surrogate models of "black-box" systems, for which we have no analytic knowledge. In the examples presented in Part II, the systems are described by commonly adopted models and the reason behind the use of the term "black-box" will become evident when we will talk about the different approaches to UQ in chapter 5.

Despite the complexity of chaos, the example presented in figures 1.1 and 1.2 is relatively simple from the UQ perspective. Simulations of the double pendulum model are very fast on today's computer architectures, and we can afford computing thousands of different solutions in a matter of minutes. Unfortunately, this is not the case for most of the models used in engineering. In Part II we will see examples where the computation of a single solution can take hours even in the most advanced architectures. Furthermore, the analysis that we sketched on the double pendulum involved only one input uncertainty. In realistic cases we often deal with tens/hundreds/thousands – sometimes infinite – number of input uncertainties. This will rapidly limit the efficiency of the proposed methods and thus a wide range of methods will be presented in Part I. In general there is no unique recipe to all the problems in UQ, but different methods need to be selected and combined in order to achieve the best outcome. In this context as in many others in engineering, the quality of the outcome is a compromise between the accuracy of the solution and the time (computational and human[5]) required to obtain it.

[5]Research usually focuses on the minimization of the computational time, while the industry, for obvious economic reasons, focuses more on the minimization of human time.

Part I

Uncertainty Quantification

Chapter 2

A formal approach to uncertainty quantification

Uncertainty Quantification (UQ) is a very active research area devoted to the study of uncertainties that affect our prediction capabilities. The problems that it addresses are based on the fields of probability theory [18, 19], dynamical systems [20–22] and numerical simulations [23–30], while the methods used are often rooted on statistics, machine learning [31, 32] and functions approximation [33, 34]. During the last two decades UQ quickly grew as an independent field and dedicated literature has appeared [35–37], providing the basis for a formal approach to UQ.

Since our prediction capabilities are bound to the concept of causality, it is obvious that the primary concern of UQ is to have – or sometimes to assume the existence – of a *model*. A model is a mathematical entity describing the causality connection between some input and some output – see figure 2.1. The models we are going to investigate here are dynamical systems, i.e. time-space dependent models, where the inputs are usually boundary conditions and/or initial values, while the outputs are the states of the system. These models, as well as their inputs, may very well depend on *parameters* $\mathbf{p}$ which affect their outputs.

In order to have any predictive capability, models need to be evaluated, associating particular input conditions to particular output states. Many models have closed form solutions which enable a fast analytic evaluation, but the majority of

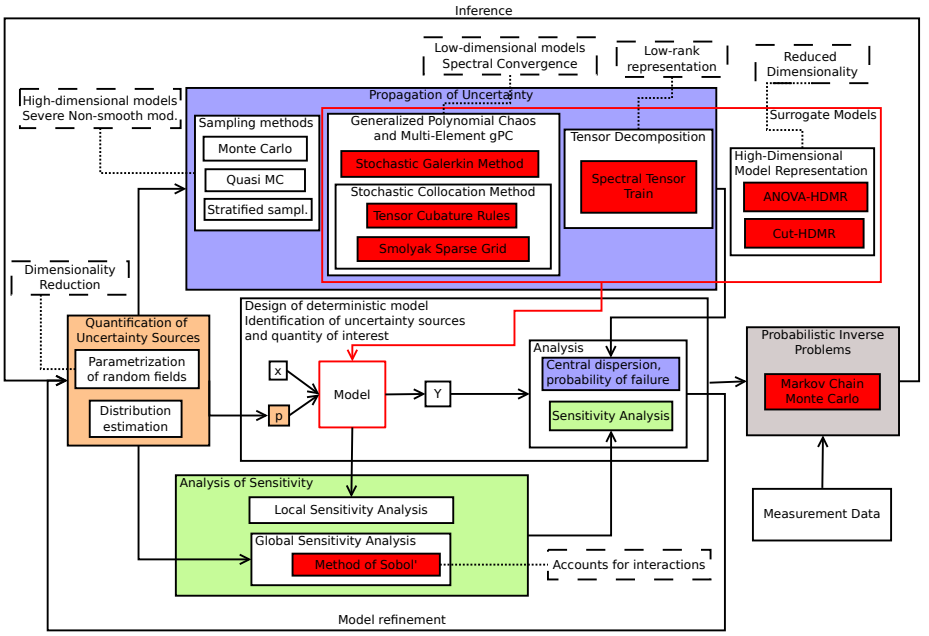


Figure 2.1: The workflow of Uncertainty Quantification. Extension of the flow chart described in [38].

them need to be simulated by complex, time consuming and ultimately expensive computational techniques. Despite the advances in numerical methods for these simulations and the evergrowing computing power, the attained incredible accuracy may very well be spoiled by the comparison with reality: the input conditions, parameters and output states are often known within "engineering accuracy", known as probability distributions or in some cases not even known. Chapter 3 will give formal definitions of dynamical systems with random inputs.

In many cases the output of dynamical systems is overabundant with respect to the analysis that we want to carry out, thus we will restrict our attention to one or more *Quantities of Interest* (QoIs) $\mathbf{Y}$. With this perspective, we redefine the model as the map between the input parameters and the output QoIs, despite the fact that the full problem might need to be computed in order to retrieve the QoIs. This map does not always need to be known analytically, but in general we need to be able to drive/measure the input, evaluate the model and measure the output, which is a minimum requirement for experiments.

Once the model, its parameters and its QoIs are defined, several kinds of analysis can be performed, not necessarily in the precise order presented here.

Types of uncertainties

Sources of uncertainties are often split in two categories: *aleatoric* and *epistemic* uncertainties [39]. The aleatoric uncertainties are the uncertainties which cannot be reduced because we do not have control on them. The epistemic uncertainties are uncertainties which could be reduced by more accurate measurements and better modeling of the physics.

This distinction is not sharp because it depends on the problem setting and on subjective judgment. Philosophically speaking, if one believes in the deterministic nature of nature – which is not the case for all the physicists –, there is no such a thing as an aleatoric uncertainty.

From the practical point of view the distinction of which uncertainty is reducible and which is not depends on the problem setting.

For example, one engineer who works on the structural safety of off-shore structures will consider the sea weather conditions to be aleatoric and can consider some parameters of the structure as epistemic. On the other hand, one of the goals of a person working on weather prediction is the reduction of the uncertainty on the forecasting and description of the sea weather conditions, which will be consider as epistemic.

Even for two different problem settings in the same engineering field, the set of aleatoric and epistemic uncertainties can differ. The most relevant example are laboratory experiments, where the explicit aim is to reduce the number of aleatoric uncertainties to a minimum. Thus, some uncertainties which are considered aleatoric in a field experiment, would be considered as epistemic in the corresponding laboratory experiment.

Even if from the practical point of view it is important to know which are the uncertainties that can be reduced, in this work we will not make any effort in their reduction. Thus, the uncertainties considered in the following will not be categorized with respect to the aleatoric and epistemic definitions.

Quantification of uncertainty sources

The uncertainty on the parameters driving the model output needs to be quantified accurately in order for the UQ analysis to be reliable. The *probability distribution estimation* of the parameters can be carried out in several ways:

- by assumption: probability distributions are assigned to the parameters relying on experience and wisdom,

- by measurement: extensive measurements of the parameters are carried out and probability distributions are fitted to these experiments,
- by inference: the probability distributions are reconstructed using measurements of the QoIs.

The quality of the UQ analysis carried out will depend strongly on the quality of the distributions constructed.

Some uncertain parameters may have a time-space dependency and this leads to the necessity of employing random processes – commonly called random fields when they are indexed by a space variable – for the description of uncertainties. Random processes are in general infinite dimensional objects, where an infinite number of length scales – in the sense of Fourier – are involved. A notorious random process is the "white noise", where all the scales contribute equally to the process. Other processes have a varying dependence on different scales and thus can be approximated by finite dimensional processes. This approximation is called *parametrization* – or sometimes dimensionality reduction – because it transforms an infinite dimensional process into one which depends only on a finite number of uncertain parameters.

Chapter 4 will present some techniques used for the probability density estimation of parameters and for parametrization random processes.

Propagation of uncertainty

Having associated probability distributions to the parameters, we want to know what the probability distributions of the QoIs look like, i.e. how the model transforms the input probabilities to the output probabilities. An extensive literature on this argument has appeared in the last 70 years, since Stanislaw Ulman came up with the Monte Carlo method in 1946 [40]. As shown in table 2.1, many methods have appeared during the years, addressing different issues. The list includes some of the methods that will be presented in chapter 5, without the ambition to present all the methods available today. As it is often the case in numerical methods, the main issues encountered in the propagation of uncertainties are related to the achievement of a good balance between accuracy and time consumed. Since "there ain't no such thing as a free lunch", all the methods presented have strengths and weaknesses, thus methods applied to particular problems need to be accurately selected.

Historically, the problem of the transformation of probability densities was first encountered in statistical mechanics in the form of the Fokker-Plank equation (1914), which is a Partial Differential Equation (PDE) describing the time evolution of the probability density function of the velocity of a particle subject to

	Name	Year
[41]	Wiener chaos expansion	1938
[40]	Monte Carlo method	1946
[42]	Quasi-Monte Carlo method	1961
[43]	Smolyak rule	1963
[44, 45]	Latin Hyper Cube	1977
[46]	Generalized Polynomial Chaos	2003
[47, 48]	Sparse Grids Quadratures	2003
[49–51]	Sparse Grid Pseudospectral approximations	2008

Table 2.1: Methods for propagation of uncertainty and approximate year of appearance.

Brownian motion forces. Later, the same problem reappeared in the dawning field of Stochastic Differential Equations (SDEs) [52] in the form of the Kolmogorov equations (1931) and the Feynman-Kac formula (1947). By all means SDEs fall into the topic of propagation of uncertainty and through Itō calculus [52, 53] they make extensive use of the techniques listed in table 2.1. This work will make use of some theory on stochastic/random processes which belongs to the theory of SDEs, but it will not delve into the solution of problems modeled by SDEs which involve Itō calculus.

Sensitivity analysis

The propagation of uncertainties is useful in describing the probability distributions of the QoIs, but it often overlooks the relation between different input uncertainties and the output uncertainty. This is the task of sensitivity analysis: explaining how the output is sensitive to the uncertainty on the input parameters.

Sensitivity analysis is also an important tool for *model refinement*: once the most influential parameters have been identified, the remaining parameters can be considered without uncertainty, leading to a model with a lower dimensional input. This refined model can then be used for other more accurate and efficient analysis.

Probabilistic inverse problems

In many practical problems the probability distribution of the input parameters is unknown and the parameters themselves are difficult, if not impossible, to be measured. In these cases we need to resort to the solution of an inverse problem,

based on the few QoIs which we can measure – sometimes called observables. In practice we wish to construct the inverse map of the model, in order to infer the inputs from some measured outputs. In the context of UQ these inverse problems are often severely under-determined, meaning that few measurements of the QoIs are available and therefore many set of parameters can produce the same results satisfying the constraints imposed by the model. Furthermore the measurements of the QoIs are often noisy, voiding even the definition of a particular set of parameters generating the output.

In this context it makes sense to rephrase the problem in terms of probability distributions: having some measurements and knowing their measurement errors try to construct probability distributions of the input parameters which are likely to have generated the available data. Solving this kind of problems is often hard, time consuming and requires a good deal of experience in the field on which they are applied. Different approaches to the problem are available and some of them will be presented in chapter 7.

Chapter 3

Dynamical systems with random inputs

> Essentially, all models are wrong, but some are useful.
>
> *George E. P. Box, 1987*

In this chapter we will blend the topics of dynamical systems and probability theory in the definition of dynamical systems with random inputs.

A review on the key concepts on dynamical systems and probability theory as well as the definition of the notation used in this work are provided in appendix A and B. The main references used are [54] on the subject of dynamical systems, [18, 19] for the subject of measure theory and probability theory, [55, 56] for subjects related to L^p spaces.

3.1 Dynamical systems

This work will consider dynamical systems in the form of *Differential Equations* (DE). The generic form of DE that we will use is an *Initial Value Problem* (IVP)

for a n-th order *Partial Differential Equation* (PDE):

$$\begin{cases} \mathbf{u}_t = \mathcal{G}\mathbf{u} & (t, \mathbf{x}) \in T \times D \\ \mathcal{B}\mathbf{u} = 0 & (t, \mathbf{x}) \in T \times \partial D \\ \mathbf{u} = \mathbf{u}_0 & (t, \mathbf{x}) \in T = t_0 \times D \end{cases} \tag{A.10}$$

where $\mathcal{G}$ is a differential operator, $\mathcal{B}$ is a boundary differential operator and $\mathbf{u}_0$ are the initial conditions for the solution $\mathbf{u} \in \mathcal{C}^n(T \times D, \mathbb{R}^m)$. Both $\mathcal{G}$ and $\mathcal{B}$ can be non-linear.

We will consider the *Boundary Value Problem* (BVP) of a PDE to be a particular case of (A.10) where the time dependency is disregarded and $\mathcal{G}\mathbf{u} = 0$. In the same way we will consider an IVP of an *Ordinary Differential Equation* (ODE) to be a particular case of (A.10) where we disregard D.

3.1.1 Numerical solution of dynamical systems

Analytic solutions for ODEs and PDEs can be found for a limited number of particular problems, where both the differential operator and the boundary conditions are well behaved – linear – and the geometry of the domain is simple. In most of the practical problems solved in engineering, the analytical solution of ODEs and PDEs is cumbersome and approximations are the only achievable results.

The last 60 years have seen the advent of computer aided simulations, with the introduction of efficient algorithms and rapidly growing computational power. Recent architectural developments pushed toward the use of massively parallel architectures and many scalable algorithms for tackling different problems have been introduced in the last decade [57, 58].

The two main approaches to the solution of the problem (A.10) are the collocation and the Galerkin methods – also known as the modal or spectral methods. The two approaches are related to each other and ultimately built upon the same approximation theory based on the Sobolev spaces to which the solution is assumed to belong.

In the collocation approach (A.10) is discretized in time and space by the introduction of the sets of points $\{t_i\}_{i=0}^{I} \in T$ and $\{\mathbf{x}_i\}_{i=0}^{N} \in D$ and by the discretization of the differential operators $\mathcal{G}$ and $\mathcal{B}$. Different selections of points and discretizations of the operators lead to different methods for solving (A.10), which however is pointwise numerically solved. The values of $\mathbf{u} \in \mathcal{C}^n(T \times D, \mathbb{R}^m)$ at the N spatially discretized points are the $m \times N$ *degrees of freedom* (d.o.f.) of the system.

In the Galerkin approach the spatial part of problem (A.10) is rewritten in terms of M modes (basis functions) which are then evolved in time accordingly with the discretized modal operators $\mathcal{G}$ and $\mathcal{B}$. In many cases, where the solutions have a sufficient regularity, the number of modes M which need to be evolved is significantly lower than N, leading to a reduced number of degrees of freedom.

The number of degrees of freedom is nowadays a milder problem than it was in the past, because the introduction of parallel computing and the development of scalable algorithms, which complexity grows almost linearly with respect to the dofs, allows to tackle bigger problems by linearly adding computational resources[1]. The other bottleneck in the solution of (A.10) is problem dependent and regards the nature of the discretized integral operator $\mathcal{G}$, which limits the convergence of numerical linear solvers or the step-size of numerical time integrators.

In general there is no recipe for all the problems, but a combination of different techniques is usually employed in order to solve large-scale problems. A deeper analysis of these numerical methods is out of the scope of this work, thus we refer the interested reader to one of the many books on the topic [23–28].

In chapter 5 we will meet again the collocation and the Galerkin approaches in the context of Uncertainty Quantification. By all means these two approaches to UQ share much of the theory with their counterparts used in numerical methods for PDEs.

3.2 Differential equations with random inputs

The operators defining dynamical systems are very often characterized by parameters which are known with a certain degree of uncertainty. We can take as a simple example (A.10), representing the heat equation with Dirichlet boundary conditions:

$$\begin{cases} \mathbf{u}_t = \nabla \cdot (\kappa(\mathbf{x}) \nabla \mathbf{u}) & (t, \mathbf{x}) \in T \times D \\ \mathbf{u} = g(\mathbf{x}) & (t, \mathbf{x}) \in T \times \partial D \\ \mathbf{u} = f(\mathbf{x}) & (t, \mathbf{x}) \in T = t_0 \times D \end{cases} \tag{3.1}$$

where κ is a space dependent thermal diffusivity. Both κ, the boundary condition g and the initial condition f can, for instance, be finite variance random fields:

$$\kappa, f \in L^2(\Omega, \mathscr{F}, P; L^\infty(D)) \,,$$
$$g \in L^2(\Omega, \mathscr{F}, P; L^\infty(\partial D)) \,.$$

[1] Often other technical problems, like bandwidth limitations, arise when using parallel resources, but architectural improvements have been and are being continuously introduced.

Consequently the solution $\mathbf{u}$ will be a random field in $L^2(\Omega, \mathscr{F}, P; L^\infty(T \times D))$. Then the system (3.1) will be rewritten as

$$\begin{cases} \mathbf{u}_t = \nabla \cdot (\kappa(\mathbf{x}, \omega) \nabla \mathbf{u}) & (t, \mathbf{x}, \omega) \in T \times D \times \Omega \\ \mathbf{u} = g(\mathbf{x}, \omega) & (t, \mathbf{x}, \omega) \in T \times \partial D \times \Omega \\ \mathbf{u} = f(\mathbf{x}, \omega) & (t, \mathbf{x}, \omega) \in T = t_0 \times D \times \Omega \end{cases} \tag{3.2}$$

In general we will rewrite the IVP (A.10) as a IVP with random inputs:

$$\begin{cases} \mathbf{u}_t = \mathcal{G}(\omega)\mathbf{u} & (t, \mathbf{x}, \omega) \in T \times D \times \Omega \\ \mathcal{B}(\omega)\mathbf{u} = 0 & (t, \mathbf{x}, \omega) \in T \times \partial D \times \Omega \\ \mathbf{u} = \mathbf{u}_0(\omega) & (t, \mathbf{x}, \omega) \in T = t_0 \times D \times \Omega \end{cases} \tag{3.3}$$

3.3 Identification of the Quantities of Interest

Often the full result $\mathbf{u}$ of an ODE or PDE is overabundant for analysis purposes. For example, consider the structural load of ocean water waves on offshore structures: the full dynamics of the water waves need to be computed in order to get reliable structural loads, but they will not be used for analysis.

In this perspective we define the functional

$$g : \Omega \to \mathbb{R}^n \tag{3.4}$$

representing the relation between the probability space $(\Omega, \mathscr{F}, P)$ and n *Quantities of Interest* (QoI). The function g will be sometimes called *QoI function* and it is in practice a random variable on $(\Omega, \mathscr{F}, P)$. We will dedicate much of this work to the characterization of this random variable, which can be viewed as the model in figure 2.1.

The QoI function is practically hiding all the technicalities of the underlying problem, from the DE model to its numerical solver. As a trivial example consider the heat transfer problem (3.1), where we are interested only in the temperature at a particular location $\mathbf{x}_0$ at the steady state time t_f. In this case we would use the definition $g(\omega) := \mathbf{u}(t_f, \mathbf{x}_0, \omega)$ as the QoI function.

In the following we will talk about *non-intrusive methods* as the methods which have no access to the underlying model of the QoI function, but can only query it. In this case the QoI function will be considered as a *black-box function*. On the contrary, we will talk about *intrusive methods* when the methods have access to the underlying model and can make use of this knowledge.

Chapter 4

Quantification of sources of uncertainty

We concluded the last chapter with the introduction of the dynamical system with random inputs (3.3) and the definition of the QoI function (3.4). In general it is not needed to identify $(\Omega, \mathscr{F}, P)$, since in a certain sense "mother nature provides it to us". Instead what we need to characterize are the random variables and random fields defined on this space, namely quantify the sources of uncertainty. We will proceed in two steps: first we will try to parametrize the QoI function (3.4) in terms of a set of independent random variables, then we will present some techniques for the characterization of the distributions of such random variables.

4.1 Parametrization of the uncertainty

Let us first consider the case in which the random input ω to the QoI function g is formed by the random vector $\mathbf{X} : \Omega \to S$, where $S \subset \mathbb{R}^{d_\mathrm{s}}$ is called the *parameter space*, d_s is the dimension of the parameter space, also known as the *co-dimension* of the system, and the random variables $\{\mathbf{X}_i\}_{i=1}^{d_\mathrm{s}}$ are mutually independent. The random vector $\mathbf{X}$ will have a distribution π in the sense of (B.3). The function

$$f : \mathbf{X}(\omega) \mapsto g(\omega) \tag{4.1}$$

is the *proper parametrization* of the QoI function and it has to be intended as the map between a realization $\mathbf{X}(\omega)$ of $\mathbf{X}$ and the corresponding QoI value $g(\omega)$. In the case that one has the analytic knowledge of the underlying model, the parametrization of g corresponds to the parametrization of its underlying dynamical system (3.3):

$$\begin{cases} \mathbf{u}_t = \mathcal{G}(\mathbf{X})\mathbf{u} & (t, \mathbf{x}, \mathbf{X}) \in T \times D \times S \\ \mathcal{B}(\mathbf{X})\mathbf{u} = 0 & (t, \mathbf{x}, \mathbf{X}) \in T \times \partial D \times S \\ \mathbf{u} = \mathbf{u}_0(\mathbf{X}) & (t, \mathbf{x}, \mathbf{X}) \in T = t_0 \times D \times S \end{cases} \tag{4.2}$$

We stress the fact that a proper parametrization is given by a set of independent random variables. If the random variables are dependent, we will simply talk about a *parametrization* of the QoI function. We will see in section 4.2 that the independence is a key property in order to introduce tensor product functional spaces and this will turn useful in chapter 5.

Let us first introduce what happens when the random input is a random field. In this case the parametrization needs to take into account the space-dependent structure of the field and a parametrization is possible through the Karhunen-Loève expansion [59, 60].

4.1.1 Karhunen-Loève expansion

We consider here a random field a defined on an arbitrary domain D with finite variance, i.e. $a \in L^2(\Omega, \mathscr{F}, P; L^\infty(D))$. After removing the mean of the field, we obtain $\tilde{a} = a - \mathbf{E}[a]$ with covariance function $\mathbf{C}_{\tilde{a}}$ given by

$$\mathbf{C}_{\tilde{a}}(\mathbf{x}, \mathbf{y}) = \int_\Omega \tilde{a}(\mathbf{x}, \omega)\tilde{a}(\mathbf{y}, \omega) P(\mathrm{d}\omega) \,. \tag{B.34}$$

We can then construct a compact Hermitian integral operator $\mathcal{V}_{\tilde{a}} : L^2(D) \to L^2(D)$ based on the kernel $\mathbf{C}_{\tilde{a}}(\mathbf{x}, \mathbf{y})$:

$$(\mathcal{V}_{\tilde{a}} u)(\mathbf{x}) = \int_{\mathbf{y} \in D} \mathbf{C}_{\tilde{a}}(\mathbf{x}, \mathbf{y}) u(\mathbf{y}) \, \mathrm{d}\mathbf{y} \,. \tag{4.3}$$

Since $\mathcal{V}_{\tilde{a}}$ is compact and Hermitian, its spectrum is formed by a countable set of eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_i \geq \ldots$ whose only point of accumulation is zero [55, 56] – i.e. $\lambda_i \searrow 0$. The eigenfunctions $\{\phi_i\}_{i=1} \subset L^2(D)$ corresponding to the non-zero eigenvalues $\{\lambda_i\}_{i=1}$ form an orthonormal system in $L^2(D)$. Then the random process a can be rewritten as the *Karhunen-Loève* (KL) *expansion*

$$a(\mathbf{x}, \omega) = \mathbf{E}[a] + \sum_{i=1}^{\infty} \sqrt{\lambda_i} \phi_i(\mathbf{x}) Y_i(\omega) \,, \tag{4.4}$$

where $\mathbf{E}[Y_i] = 0$ and $\mathbf{Cov}[Y_i, Y_j] = \delta_{ij}$. As we point out in appendix B.5, the uncorrelation of two random variables does not imply their independence. Thus $f : \mathbf{Y}(\omega) \mapsto g(\omega)$ is merely a parametrization of the QoI function. In the case that a is a Gaussian random field, then $\mathbf{Y}$ is a Gaussian random vector and $\mathbf{Cov}[Y_i, Y_j] = \delta_{ij}$ implies that $\{Y_i\}_{i=1}^{\infty}$ are mutually independent. This means that $Y_i \sim \mathcal{N}(0, 1)$ and f is a proper parametrization of the QoI function. In all the other cases the independence assumption is not rigorous, even if it is still used in common practice. Some techniques for finding a transformation from an independent random vector $\mathbf{Z}$ to the dependent random vector $\mathbf{Y}$ are presented in section 4.2.

The KL-expansion (4.4) needs to be truncated for practical use. Note that

$$\begin{aligned} \|\mathbf{V}[a(\mathbf{x}, \omega)]\|_{L^1(D)} &= \left\| \sum_{i,j=1}^{\infty} \sqrt{\lambda_i \lambda_j} \phi_i \phi_j \mathbf{E}[Y_i Y_j] \right\|_{L^1(D)} \\ &= \sum_{i=1}^{\infty} \lambda_i \left\| \phi_i^2 \right\|_{L^1(D)} = \sum_{i=1}^{\infty} \lambda_i \ . \end{aligned} \tag{4.5}$$

Then we can select $N_{KL} \in \mathbb{N}_+$ such that

$$q \leq \frac{\sum_{i=1}^{N_{KL}} \lambda_i}{\|\mathbf{V}[a(\mathbf{x}, \omega)]\|_{L^1(D)}} \ , \tag{4.6}$$

where $0 < q < 1$ defines the portion of variance that will be represented by the *KL-approximation*

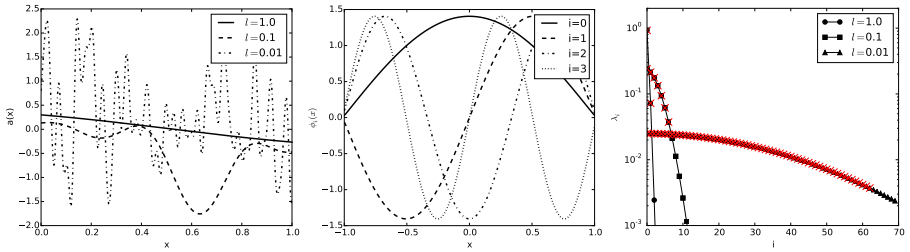
$$\hat{a}(\mathbf{x}, \omega) = \mathbf{E}[a] + \sum_{i=1}^{N_{KL}} \sqrt{\lambda_i} \phi_i(\mathbf{x}) Y_i(\omega) \simeq a(\mathbf{x}, \omega) \ . \tag{4.7}$$

Figure 4.1 shows an example of KL-approximation applied to a one-dimensional Gaussian random field with squared exponential covariance. We can see that the field becomes rougher as the correlation length becomes smaller and at the same time the number N_{KL} of retained components in the KL-approximation increases for the same level of expressed variance.

The covariance $\mathbf{C}_{\tilde{a}}$ is said to be separable if for $D = \prod_{i=1}^{d} D_i$, there exist the covariances $C_i : D_i \times D_i \to \mathbb{R}$ such that

$$\mathbf{C}_{\tilde{a}}(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{d} C_i(x_i, y_i) \ . \tag{4.8}$$

In this case the KL-expansion can be separately applied to each of the covariances C_i and then combined via tensor product.



(a) Realizations (b) Eigenfunctions $l = 0.01$ (c) Eigenvalues

Figure 4.1: Example of KL-expansion on a Gaussian random field with squared exponential covariance. Left: realizations with different correlation lengths l. Center: first four eigenfunctions for $l = 0.01$. Right: decay of the eigenvalues for different correlation lengths l. The crossed eigenvalues are the one retained in the KL-approximation (4.7) in order to represent $q = 95\%$ of the variance.

Furthermore, we refer the reader to [61] for the case of vector-valued dependent random fields, which need a special treatment in order for the approximation error of each vector component to be correctly weighted.

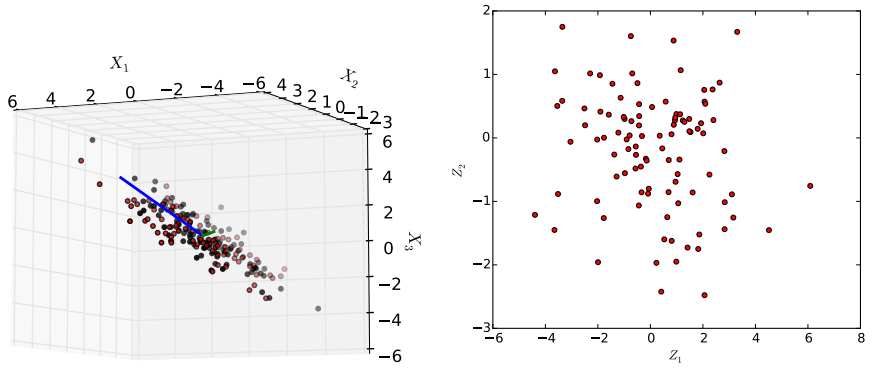
The KL-expansion and its truncation strategy belong to the methods for *dimensionality reduction*. In general, given some functional $f \in L^p$, these methods find a subspace in L^p which lead to the minimum error in the corresponding L^p norm. In the particular case of the KL-expansion (4.4), we see that $a \in L^2(D \times \Omega)$ with $\dim(L^2(D \times \Omega)) = \infty$, and the truncation strategy (4.6)-(4.7) allows the selection of the N_{KL}-dimensional subspace such that

$$S_{N_{KL}} = \underset{\substack{B<L^2(D\times\Omega)\\ \dim(B)=N_{KL}}}{\arg\min} \|a - \mathcal{P}_B a\|_{L^2(D\times\Omega)} \tag{4.9}$$

where $B < L^2(D \times \Omega)$ means that B is a subspace of $L^2(D \times \Omega)$ and $\mathcal{P}_B : L^2(D \times \Omega) \to B$ is the projection operator onto B. We will meet this technique often along this work.

4.2 Independence of random vectors

In the next chapter we will aim at the characterization of the distribution of f – c.f. (4.1) –, and we will need to query f at some particular values of $\mathbf{X}$. Doing so requires drawing some realizations – *sampling* – $\{\mathbf{X}^{(i)}(\omega)\}_{i=1}^N$ from π. However, the distribution π can be very complex for $d_s \gg 1$ and sampling from it can be problematic.



(a) Samples of $\mathbf{X}$ (b) Samples of $\mathbf{Z}$

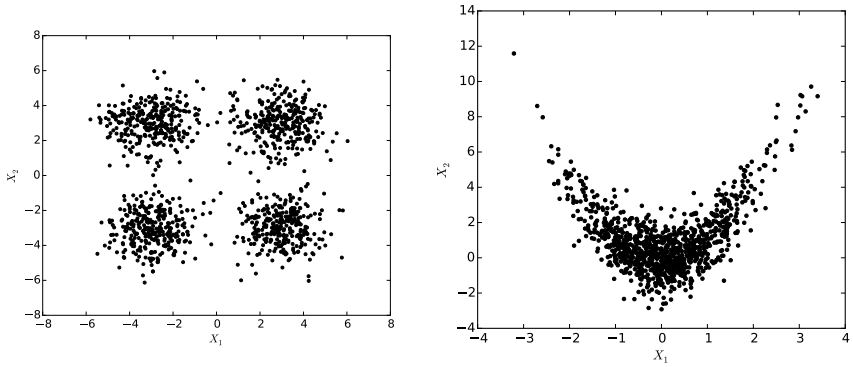
Figure 4.2: Figure 4.2a shows some samples of $\mathbf{X}$ in black, where we can see that the distribution clusters on a two-dimensional subspace of $\mathbb{R}^3$. Finding this subspace allows us to sample from the two dimensional random vector $\mathbf{Z}$ as shown in figure 4.2b and to project back these points in $\mathbb{R}^3$, occurring in the minimal error with respect to the norm on $L^2_\pi(\mathbb{R}^3)$.

In many situations one can assume that $\mathbf{X} = \{X_1, \ldots, X_{d_s}\}$ is formed by independent random variables, i.e. π is a product measure: $\pi = \prod_{i=1}^{d_s} \pi_i$. This allows the introduction of a coordinate system in $S = S_1 \times \ldots \times S_{d_s}$ for which each dimension i has its own measure π_i. As a consequence, in order to sample from the d_s dimensional distribution π one needs only to sample independently from the one dimensional distributions $\{\pi_i\}_{i=1}^{d_s}$. Additionally, this means that $L^2_\pi(S) = L^2_{\pi_i}(S_i) \otimes \ldots \otimes L^2_{\pi_i}(S_i)$ is a tensor product space[1].

The independence of $\{X_i\}_{i=1}^{d_s}$ will turn out to be very useful in chapter 5, so it is reasonable to try to parametrize the QoI function in terms of independent random variables even when this cannot be assumed. Unfortunately this is not an easy task in all the situations.

When the random vector $\mathbf{X}$ has a centered Gaussian distribution, its characterization is solely given by its covariance matrix $\mathbf{C_X}$. An independent random vector would have a diagonal covariance matrix with the variances of the independent variables on its diagonal. If we let $\mathbf{X} = (X_1, \ldots, X_{d_s})$ be a random vector of dependent Gaussian variables, there is an easy linear transformation to the vector of independent Gaussian variables $\mathbf{Y} = (Y_1, \ldots, Y_{d_s})$ where $Y_i \sim \mathcal{N}(0, \sigma_i^2)$, $\mathbf{Y} \sim \mathcal{N}(\mathbf{0}, \mathbf{C_Y})$ and $\mathbf{C_Y} = \mathrm{diag}(\boldsymbol{\sigma}^2)$. The transformation matrix

[1]Note that while the construction of tensor product Hilbert spaces is well behaved because it relies on the construction of a new inner product, the construction of tensor product Banach spaces is subtle because the construction of a new norm for the tensor product space is not unique.



(a) Samples from (4.10) (b) Samples from (4.11)

Figure 4.3: Sampling from uncorrelated but dependent random variables.

A such that $\mathbf{X} = A\mathbf{Y}$ is given by the Cholesky decomposition of the covariance matrix $\mathbf{C_X} = AA^T$.

When $\mathbf{C_X}$ is singular or close to singular, it means that two random variables are strongly correlated and thus we can equivalently use a lower dimensional vector $\mathbf{Z} = (Z_1, \ldots, Z_{d_\Xi})$, where $d_\Xi < d_s$. This can be done employing the Principal Components Analysis (PCA). This consists in the computation of the eigenvalue decomposition $\mathbf{C_X}P = P\Lambda$, where $\mathbf{C_X}$ is semi-positive definite, P is a matrix of orthogonal columns containing the principal directions in $\mathbb{R}^{d_s}$ and Λ are the eigenvalues of the principal directions. The directions with the smallest eigenvalues λ_i will have a negligible influence on the total variance $\sum \mathbf{V}[X_i] = \sum \lambda_i$. Thus we can retain the d_Ξ largest eigenvalues and discard the remaining directions. The principal directions are orthogonal and define a coordinate system for $\mathbf{Z}$, where $Z_i \sim \mathcal{N}(0, \lambda_i)$. Realizations $Z = \{\mathbf{Z}^{(j)}\}_{j=1}^N$ of $\mathbf{Z}$ can then be projected back to the S space to obtain realizations $X = \{\mathbf{X}^{(j)}\}_{j=1}^N$ of $\mathbf{X}$. This corresponds to the evaluation of $X = PZ$. This procedure is shown in figure 4.2.

Unfortunately, the techniques presented up to here work only for Gaussian random vectors. In fact, as we already stated, a non-Gaussian uncorrelated random vector is not necessarily independent, thus diagonalizing the covariance matrix doesn't work for non-Gaussian vectors. For instance, in figure 4.3a $\mathbf{X}$ is an uncorrelated dependent random vector with a multi-modal Normal distribution π with PDF

$$\rho_{\mathbf{X}}(\mathbf{x}) = \frac{1}{4}\sum_{j=1}^{4} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{|\mathbf{x} - \mu_j|^2}{2}\right) , \tag{4.10}$$

where $\mu = [(3,3),(3,-3),(-3,3),(-3,-3)]$. In figure 4.3b $\mathbf{X}$ is a uncorrelated

dependent random vector:

$$X_1 \sim \mathcal{N}(0,1)\ , \quad X_2 = X_1^2 + Z\ , \tag{4.11}$$

where $Z \sim \mathcal{N}(0,1)$. For both of these cases the correlation between the two variables is zero, but they are strongly dependent by construction. For instance in the second case:

$$\mathbf{Cov}\left[X_1, X_2\right] = \mathbf{E}\left[X_1 X_2\right] - \mathbf{E}\left[X_1\right]\mathbf{E}\left[X_2\right] = \mathbf{E}\left[X_1^3\right] + \mathbf{E}\left[X_1 Z\right] = 0\ .$$

In these cases what one needs to know are the relations between the variables, i.e. the conditional CDF's. This is the idea behind the Rosenblatt transformation [62]. If we let $\mathbf{X} = (X_1, \ldots, X_{d_\mathrm{s}}) \sim \pi$ be a vector of dependent random variables, we can define $\mathbf{Z} = (Z_1, \ldots, Z_{d_\mathrm{s}})$ by

$$\begin{aligned} Z_1 &\sim \pi_{X_1}\ , \\ Z_2 &\sim \pi_{X_2|X_1}\ , \\ &\ldots \\ Z_{d_\mathrm{s}} &\sim \pi_{X_{d_\mathrm{s}}|X_{d_\mathrm{s}-1},\ldots,X_1}\ . \end{aligned} \tag{4.12}$$

It can be shown that $\mathbf{Z}$ is a vector of independent random variables. The drawback of this approach is that it requires the knowledge of these conditional probabilities which are rarely available in practice.

The Rosenblatt transformation belongs to the class of *indirect methods* for proper parametrization. Given $\mathbf{X} : \Omega \to S \subset \mathbb{R}^{d_\mathrm{s}}$ with distribution $\pi_{\mathbf{X}}$, the aim of these methods is the construction of a function $t : \boldsymbol{\Xi} \to S$ such that $t(\mathbf{Z}) \sim \pi_{\mathbf{X}}$, where $\mathbf{Z} : \Omega \to \boldsymbol{\Xi} \subset \mathbb{R}^{d_{\boldsymbol{\Xi}}}$ is an independent random vector with a known distribution $\pi_{\mathbf{Z}} = \prod_{i=1}^{d_{\boldsymbol{\Xi}}} \pi_i$ and $d_{\boldsymbol{\Xi}} \leq d_\mathrm{s}$. This can be achieved by the solution of a probabilistic inverse problem for some parametric formulation of t, with respect to some collected measurement.

Probabilistic inverse problems will be the topic of section 7, but they will be defined in a slightly different context where the inference of the distribution of the input parameters is done with respect to measurements of the output QoIs rather than the input parameters themselves.

Other methods aim at the direct construction of the probability density function and thus belong to the class of *direct methods*. In the next section we will see some basic examples of this approach.

The interested reader is referred to [63–67] for more advanced direct and indirect methods.

4.3 Probability density estimation

Most of the distributions used in practical applications of probability theory admit densities – c.f. (B.5). It is thus useful to identify these densities because they completely characterize the distribution and can then be used to draw realizations, e.g. by rejection sampling, for the propagation of uncertainty. The identification of the densities goes under the name *probability density estimation* and consists in the construction of densities which agree with measurement data of the random vector of interest – at this stage the random vector of input uncertainties.

We can distinguish between two classes of methods for probability density estimation: the *parametric methods* and the *non-parametric methods*.

4.3.1 Parametric methods

The *parametric methods* are based on the *a priori* selection of a family of probability distributions $\{\pi_\theta : \theta \in \Theta\}$ for $\mathbf{X}$, parametrized by a finite set of parameters θ – e.g. the family of Normal distributions is parametrized by its mean and variance – and a sufficient statistic which can completely determine those parameters – e.g. for the family of Normal distributions the sufficient statistic is the sample mean and the sample variance. In practical cases the family of probability distributions can be formed by mixtures of multiple distributions. In this case the unknown parameters are both the parameters of each distributions and the mixing coefficients, called latent variables.

The problem of determining θ given a set of observations $\{\mathbf{x}_i\}_{i=1}^N$ is commonly rephrased into a *Maximum Likelihood* optimization problem:

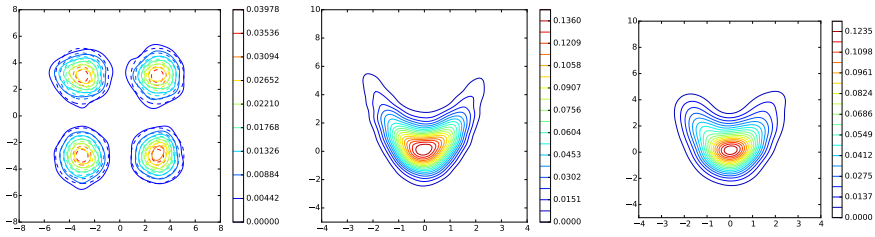
$$\hat{\theta} = \arg\max_{\theta \in \Theta} \ell\left(\theta; \{\mathbf{x}_i\}_{i=1}^N\right) \tag{4.13}$$

where ℓ is the log-likelihood

$$\ell\left(\theta; \{\mathbf{x}_i\}_{i=1}^N\right) = \log\left(\prod_{i=1}^N \rho_\theta(\mathbf{x}_i)\right) = \sum_{i=1}^N \log \rho_\theta(\mathbf{x}_i) \tag{4.14}$$

and ρ_θ is the PDF of π_θ. The latest form of the log-likelihood in (4.14) is usually preferred to the central one, because of the risk of arithmetic underflow when a product of small values is taken.

In a limited set of cases this problem has a closed form and can be solved by simple linear algebra. In general, however, the maximizer must be identified



(a) PDF of (4.10) – $\lambda = 0.5$ (b) PDF of (4.11) – $\lambda = 0.3$ (c) PDF of (4.11) – $\lambda = 0.5$

Figure 4.4: Kernel Density Estimation of examples in figure 4.3. Estimation obtained using the kernel (4.17), and different smoothing parameter λ. On the left figure, the dashed curves show the exact PDF.

numerically. If the optimization problem can be shown to admit only one maximizer, then a deterministic optimization algorithm can be used. In the most general case the problem can have multiple maximizers, thus one need to resort to random search algorithms. Alternatively, a Bayesian model of the form

$$\rho^{\text{post}}(\theta|\{\mathbf{x}_i\}_{i=1}^N) \propto \ell\left(\theta; \{\mathbf{x}_i\}_{i=1}^N\right) \rho^{\text{prior}}(\theta) \tag{4.15}$$

can be set up. The posterior distribution of the parameters $\rho^{\text{post}}(\theta|\{\mathbf{x}_i\}_{i=1}^N)$ can be sampled as described in chapter 7, and the most probable parameter set θ can be taken as the maximizer of problem (4.13). See [32] for further details on these methods.

Once the parameters θ have been identified, the analytic knowledge of the distribution can be exploited in the forward propagation of uncertainty, by direct sampling or by the construction of more advanced methods – see chapter 5.

4.3.2 Non-parametric methods

On the other hand the *non-parametric methods* do not make any assumption regarding the family of probability distribution to which $\mathbf{X}$ belongs. They try instead to construct the PDF ρ of the d_{s}-dimensional random vector $\mathbf{X} \sim \pi$ only from the available data $\{\mathbf{x}_i\}_{i=1}^N$, using the *Kernel Density Estimation* (KDE):

$$\rho(\mathbf{x}_0) = \frac{1}{N} \sum_{i=1}^{N} K(\mathbf{x}_0, \mathbf{x}_i) \, . \tag{4.16}$$

A common choice of K is the Gaussian kernel density:

$$K(\mathbf{x}_0, \mathbf{x}) = \frac{1}{(2\lambda^2\pi)^{\frac{d_{\text{s}}}{2}}} \exp\left(-\frac{\|\mathbf{x}_0 - \mathbf{x}\|^2}{2\lambda^2}\right) \, , \tag{4.17}$$

where λ, the variance, plays the role of a scaling factor which defines how quickly the relevance carried by one measurement decay as we get farther from it. In practice λ provides a tunable smoothing parameter for the approximation. With $\lambda \to 0$, the distribution ρ will tend to a sum of N Dirac distributions centered at the measurements. For multiple dimensions, (4.17) can be rewritten in order to have anisotropic variance. In literature λ is commonly called "bandwidth" and the reader is referred to [68] for a review on its automatic selection.

Figure 4.4 shows the KDE applied to the examples (4.10)-(4.11) using the samples in figure 4.3. We notice that the smoothing parameter λ gives an important contribution to the quality of the estimated PDF. A drawback of KDE is that the number of measurements needs to be big in order to obtain reliable approximations, in particular in high-dimension, i.e. for $d_{\mathrm{s}} \gg 1$.

Once the PDF has been estimated, it can be used for the forward propagation of uncertainty. However, the lack of analytical knowledge of the associated distribution, leads to the need of using algorithms such as rejection-sampling, which are not efficient in high-dimension and for concentrated distributions, or a Bayesian approach using Markov Chain Monte Carlo – see chapter 7.

Chapter 5

Propagation of uncertainty

In the last chapter we described the construction of the parametrization

$$f : \mathbf{X}(\omega) \mapsto f(\omega) \tag{4.1}$$

of the QoI function g described in section 3.3, in terms of the random vector $\mathbf{X} \sim \pi_{\mathbf{x}}$. In section 4.2 we stressed that if f is a proper parametrization, i.e. if the components of $\mathbf{X}$ are mutually independent, we would gain many advantages in the propagation of the uncertainty. However, not all the methods presented here need a proper parametrization in order to work, but in general any of them would benefit from having it.

In this chapter we will present methods for the analysis of the random variable $f \circ \mathbf{X} : \Omega \to \mathbb{R}^n$. We will use the shorthand π_f for the distribution $\pi_{f\circ\mathbf{x}}$ defined for any $A \in \mathscr{B}(\mathbb{R}^n)$ as

$$\pi_{f\circ\mathbf{x}}(A) = \pi_{\mathbf{x}}\left(f^{-1}(A)\right) = P\left(\mathbf{X}^{-1}\left(f^{-1}(A)\right)\right) \ , \tag{5.1}$$

where $\mathbf{X}^{-1}$ and f^{-1} are the pre-images of $\mathbf{X}$ and f respectively.

From the UQ perspective the analysis of the random variable $f \circ \mathbf{X}$ can mean several things:

- **Statistics**: we could be interested to the statistical moments of f, e.g. on its average behavior $\mathbf{E}[f]$ and its variance $\mathbf{V}[f]$.

- **Distributions**: since $f \circ \mathbf{X}$ is very often a non-Gaussian random variable, the first statistical moments do not provide sufficient information regarding its distribution π_f, thus we could aim at its approximation.

- **Probabilities**: In some cases we are not interested on the general distribution π_f, but only on the probability of some event to occur, i.e. on the measure of this event under π_f.

The particular analysis that is of interest is problem dependent and here we will focus mainly on the analysis of statistics and distributions. The analysis of probabilities is usually addressed in the context of *reliability analysis*, where the events of interest are extreme events with, hopefully, low probability. The reader is referred to [69–71] for more information regarding the methods for this kind of analysis.

The methods presented here will make different levels of assumptions. Assumptions usually help improving the performances of the propagation of uncertainty, but they also limit the applicability of the methods. Here we list the main assumptions which will be mentioned later on.

(PU-0) Any desired value can be assigned to the random vector $\mathbf{X}$. In other words, one is able to drive the input of the QoI function f.

(PU-1) f is a proper parametrization, i.e. $\mathbf{X} \sim \pi_{\mathbf{x}}$ is composed by mutually independent random variables and $S = \prod_{i=1}^{d_s} S_i$, with $S_i \subset \mathbb{R}$,

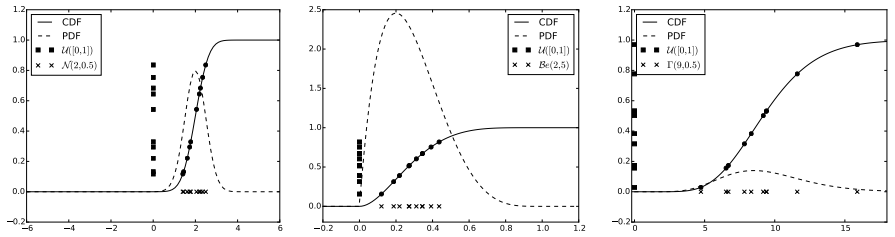
(PU-2) $f \in L^2_{\pi_{\mathbf{x}}}(S)$,

(PU-3) $f \in \mathcal{H}^k_{\pi_{\mathbf{x}}}(S)$ for some $k \geq 0$, i.e. f possess a certain degree of regularity with respect to $\mathbf{X}$.

(PU-4) One has explicit knowledge of f and of the underlying dynamical system, including its implementation.

Even if assumption (PU-0) seem to be obviously always fulfilled, there are situations where this is not the case. Consider for example the input data of a laboratory experiment where $\mathbf{X}$ could only be measured but not controlled.

In the discussion of this chapter we will disregard analytical methods for the calculation of the statistics and the transformation of distributions, due to their limited applicability. We will instead focus on numerical methods which are generally applicable to any problem in the form (4.1).



(a) $\mathcal{N}(2, 0.5)$ (b) $\mathcal{B}e(2, 5)$ (c) $\Gamma(9, 0.5)$

Figure 5.1: Examples of the inverse sampling. The inverse CDF function is evaluated on values drawn from the uniform distribution $\mathcal{U}([0, 1])$ (square dots along the vertical direction), obtaining values with the desired distribution (crosses along the horizontal direction).

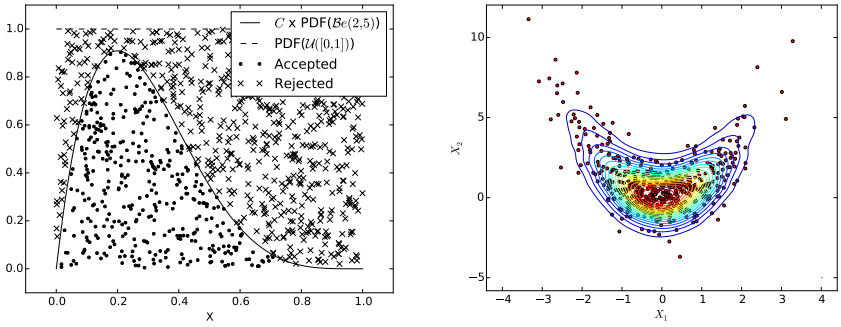
5.1 Pseudo-random sampling methods

Pseudo-random sampling methods are the most general methods for propagation of uncertainty, making very little assumptions on the QoI function and its parametrization. These methods are frequently referred to as brute force methods, because they practically try to mimic the probabilistic characteristics of nature, which here is described by the probability space $(\Omega, \mathscr{F}, P)$.

Before introducing the pseudo-random sampling methods, we need to clarify what pseudo-random sampling means.

Given a random vector $\mathbf{X} \sim \pi$ – with not necessarily mutually independent components – *random sampling* means to draw an *ensemble* $\{\mathbf{X}^{(i)}(\omega)\}_{i=1}^{N}$ formed by N realizations from $\{\mathbf{X}^{(i)}\}_{i=1}^{N}$, which are independent and identically distributed (i.i.d.) random vectors with $\mathbf{X}^{(i)} \sim \pi$. Random sampling is impossible in practice. Thus we always resort to *pseudo-random sampling* a sequence of values $\{\mathbf{X}^{(i)}(\omega)\}_{i=1}^{N}$ which are selected according to π and not correlated to each other. Since the generating algorithm is deterministic, the latter property is not fulfilled in practice. However, the correlation length of the values generated by nowadays *pseudo-random number generators* (RNG) is usually much longer than the size N of the desired sample set, and thus it is never a problem. The reader is referred to [72, 73] for more details on the practical implementation of pseudo-random number generators.

All the basic algorithms for pseudo-random number generation try to sample from a uniform distribution $\mathcal{U}([0, 1))$ and thus we are in principle only able to generate values between 0 and 1. In order to sample from a more complex distribution π, we need to use a mapping from the uniform distribution to the



(a) $\mathcal{B}e(2,5)$ (b) KDE of (4.11)

Figure 5.2: Examples of rejection sampling.

distribution of interest. For example let $X \sim \pi$ be the random variable of interest from which we want to sample and let F_X be its CDF. From (B.4), F_X is continuous and non-decreasing and thus we can define its left-continuous inverse as

$$F_X^{-1}(u) = \inf\{x : F_X(x) \geq u\} \,. \tag{5.2}$$

With this definition and for $U = F_X(X)$ we have that

$$P(U \leq u) = P(F_X(X) \leq u) = P(X \leq F_X^{-1}(u)) = F_X(F_X^{-1}(u)) = u$$

and thus $U \sim \mathcal{U}([0,1))$. This implies that $F_X^{-1}(U) \sim \pi$. Thus, the application of this transformation to a uniformly sampled variable, leads to the sampling from the distribution of interest. This method is called *inverse sampling* and figure 5.1 shows its application for the generation of samples with different distributions. Most of the software packages devoted to RNG provide samplers of commonly used distributions implemented with the inverse sampling method.

In spite of being very efficient in the generation of pseudo-random numbers, the applicability of the inverse sampling method is limited to one dimensional distributions – this excludes dependent random vectors – and distributions for which the inverse transform is known.

An alternative method of sampling, which only requires the knowledge of the PDF ρ_X of the random vector $\mathbf{X}$, is the *rejection sampling*. The idea is to sample from a known distribution ρ_Y which dominates almost everywhere $C\rho_X$ for a properly selected $0 < C < 1$. For example, let $X \sim \mathcal{B}e(2,5)$ and let us assume that we know its density ρ_X its inverse CDF is unknown. We let $Y \sim \mathcal{U}([0,1])$ and fix $C = 1.1 \times \max(\rho_X)$ as shown in figure 5.2a. Then we can exploit the following property: if (X,Y) is uniformly distributed in $\{(x,y) : 0 \leq y \leq \rho_X(x)\}$, then the PDF of X is ρ_X. Note that sampling uniformly

under the dominating PDF ρ_Y leads to sampling uniformly under the dominated distribution ρ_X. Thus we can uniformly sample under the graph of ρ_Y and reject the samples which are above the graph of $C\rho_X$. This strategy is shown in figure 5.2a for the Beta distribution. Figure 5.2b shows the more involving case regarding the dependent random vector (4.11) for which the PDF was approximated using the KDE method in section 4.3.2.

Rejection sampling is a powerful method to be used when the PDF is known and can be easily dominated by the PDF of a distribution from which we are able to sample. The best dominating distribution is the one for which the volume of the gap between its PDF and the PDF of interest is minimum, because in this way one will accept the highest number of samples. If the PDF of interest does not resemble any of the analytically known distributions – e.g. if it has peaks –, then rejection sampling will end up selecting many values that will be rejected. This problem quickly amplifies as the dimension increases, because the rejection volume grows exponentially with the dimension, while the acceptance volume is fixed. Sampling from high-dimensional distributions is usually addressed using one of the Markov Chain Monte Carlo methods presented in chapter 7.

5.1.1 Monte Carlo method

> Monte Carlo is an extremely bad method, it should be used only when all alternative methods are worse.[1]
>
> *Alan Sokal, 1996*

The aim of the Monte Carlo (MC) method is to sample from the distribution π_f of the parametrized QoI function f (4.1), and to characterize π_f through the collection of realizations $\{(f \circ \mathbf{X})^{(i)}\}_{i=1}^N$.

This is by far the most generic method among the ones for propagation of uncertainty. Assumption (PU-0) is generally required, but in some cases it can also be neglected. This assumption is usually not fulfilled when performing laboratory experiments, where the input $\mathbf{X} \sim \pi$ cannot be controlled but only measured. However, if the laboratory experiment guaranties that the realizations of $\mathbf{X}$ are actually distributed with distribution π, then the assumption (PU-0) is not even necessary. In this case we would refer to MC as a random sampling method instead of a pseudo-random sampling method.

[1] Which is often the case.

Given a parametrized QoI function f, $\mathbf{X} \sim \pi_X$ and an RNG for π_X, the method can be summarized in three steps:

1. Use the RNG to obtain the set $\{\mathbf{X}^{(j)}(\omega)\}_{j=1}^N$ of N realizations of the i.i.d. random vector $\{\mathbf{X}^{(j)}\}_{j=1}^N$.
2. Compute the set $\{(f \circ \mathbf{X})^{(i)}(\omega)\}_{i=1}^N$, where $(f \circ \mathbf{X})^{(j)}(\omega) = f(\mathbf{X}^{(j)}(\omega))$. In other word this corresponds to the evaluation of f on $\{\mathbf{X}^{(j)}(\omega)\}_{j=1}^N$
3. Use $\{(f \circ \mathbf{X})^{(i)}(\omega)\}_{i=1}^N$ to characterize π_f. This can mean the computation of the relevant statistics of π_f or the characterization of π_f through some of the methods presented in section 4.3[2].

The method hinges on the definition of sample mean

$$\mathbf{E}[f] = \mu_f \simeq \bar{\mu}_f = \frac{1}{N} \sum_{i=1}^{N} f(\mathbf{X}^{(j)}) \tag{5.3}$$

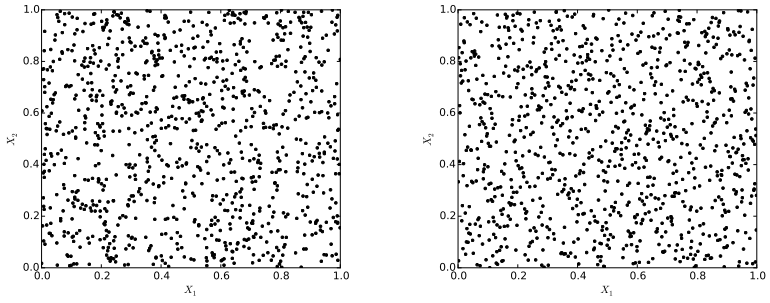
and on the Central Limit Theorem, which implies that $\bar{\mu}_f \to \mathcal{N}(\mu_f, \sigma_f^2/N)$ as $N \to \infty$. This means that the standard deviation of the mean estimator (5.3), which broadly speaking represents the error, decreases with the inverse of the square root of N, i.e. $\mathcal{O}(N^{-1/2})$.

This convergence rate is rather slow compared to convergence rates that we are used to encounter for numerical algorithms. This poses a big limitation to the applicability of the method and its accuracy when the evaluation of f is computationally expensive. However, the method is widely used mainly due to its robustness and ease of implementation. But also for a more critical property: the estimator (5.3) has a convergence rate of $\mathcal{O}(N^{-1/2})$ independently from the dimension d_s of $\mathbf{X}$. This is a very useful property because many random variables are often involved by f, and no other method possesses this property unless other assumptions are made on f. Methods for which the convergence rate deteriorates with the increase of the dimension d_s are said to suffer the *curse of dimensionality*

5.1.2 Latin Hyper Cube

The MC method uses the RNG to produce the ensemble $\{\mathbf{X}^{(j)}(\omega)\}_{j=1}^N$ which is then used as an argument for f. As we discussed in the last section, the

[2]Section 4.3 describes the characterization of source of uncertainty. In this case however we would use those methods for the characterization of the uncertainties of the QoI rather than the sources. We see here an example of how the flowchart of the UQ workflow in figure 2.1 is a simplification and methods belonging to different UQ stages can be used in other stages as well.



(a) Monte Carlo (b) Latin Hyper Cube

Figure 5.3: Monte Carlo and Latin Hyper Cube samples of $\mathbf{X} \sim \mathcal{U}([0,1]^2)$

convergence of the estimated moments of π_f to their exact counterparts is very slow. This is partly due to the fact that the quality of the ensemble $\{\mathbf{X}^{(j)}(\omega)\}_{j=1}^N$ is poor and it converges very slowly to the desired distribution π_X. Latin Hyper Cube (LHC) [44] helps improving the quality of the ensemble and then it often leads to a faster convergence of the estimators. Convergence rates are difficult to obtain for this method because they are problem dependent, but an estimate is given in [44].

But for the sampling strategy of the ensemble $\{\mathbf{X}^{(j)}(\omega)\}_{j=1}^N$, the LHC method follows the same three steps of the MC method. Assuming that $\mathbf{X} \sim \mathcal{U}([0,1]^{d_\mathrm{s}})$, the ensemble $\{\mathbf{X}^{(j)}(\omega)\}_{j=1}^N$ is constructed as follows:

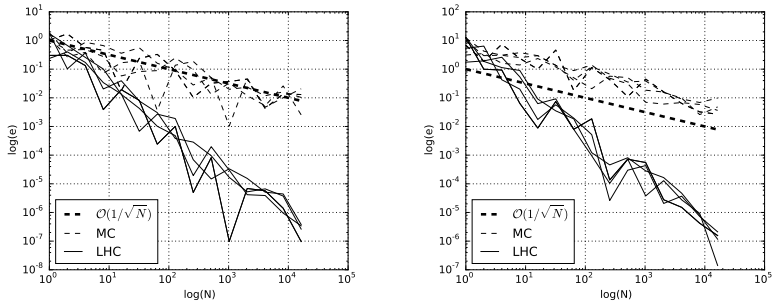
1. each of the d_s axis are partitioned into N parts, creating N^{d_s} cells $\{c_\mathbf{i}\}_{\mathbf{i}=\mathbf{1}}^{\mathbf{N}}$,
2. the N samples are extracted such that each of them belongs to a cell $c_\mathbf{i}$ and no other sample is contained on cells which share indices with $\mathbf{i}$,

where $\mathbf{i} = (i_1, \ldots, ß_{d_\mathrm{s}})$ is a multi-index and the notation $\{\cdot\}_{\mathbf{i}=\mathbf{1}}^{\mathbf{N}}$ means for all $\mathbf{i} \in \{\mathbf{i} : 1 \leq i_j \leq N_j,\ \forall j \in [1, \ldots, d_\mathrm{s}]\}$. The second step seems difficult to be enforced, because the number of cells grows exponentially with d_s. However, with a small implementation trick an algorithm with $\mathcal{O}(d_\mathrm{s} N)$ can be obtained[3].

As usual, once that we are able to sample from the uniform distribution, we can generate ensembles from other distributions using, for instance, the inverse transform method.

Figure 5.4 shows the convergences of the Monte Carlo method and the Latin Hyper Cube methods on the estimation of the mean of the function $f(\mathbf{X}) =$

[3]For example, see the implementation in [Bigoni, 12]



(a) $d_\mathrm{s} = 2$ (b) $d_\mathrm{s} = 100$

Figure 5.4: Convergence of Monte Carlo and Latin Hyper Cube mean estimator to $\mathbf{E}\left[f(\mathbf{X})\right] = \mathbf{E}\left[3\sum_{i=1}^{d_\mathrm{s}} \mathbf{X}_i^2\right] = d_\mathrm{s}$ with $\mathbf{X} \sim \mathcal{U}([0,1]^{d_\mathrm{s}})$

$3\sum_{i=1}^{d_\mathrm{s}} \mathbf{X}_i^2$, where $\mathbf{X} \sim \mathcal{U}([0,1]^{d_\mathrm{s}})$. We can see that the LHC method significantly outperform the MC method, which exhibits the usual convergence $\mathcal{O}(1/\sqrt{N})$. It is also interesting to note that none of the two methods suffer from the curse of dimensionality.

An improvement of the LHC method is the Orthogonal LHC method [74], which is able to construct ensembles of even better quality. A totally different approach is instead taken by the Quasi-Monte Carlo (QMC) method [42], which is based on deterministic sequences and their randomization, and leads to a convergence of $\mathcal{O}\left((\log N)^{d_\mathrm{s}}/N\right)$ which is asymptotically better than MC and LHC, but it deteriorates with increasing d_s.

5.2 Polynomial chaos methods

The pseudo-random sampling methods presented up to here are mainly designed to compute the moments of the random variable $f \circ \mathbf{X}$. The dependence relation between f and its input $\mathbf{X} : \Omega \to S \subset \mathbb{R}^{d_\mathrm{s}}$ is mostly lost using these methods. Polynomial chaos (PC) methods try to exploit this relation in order to get better estimates with a lower computational burden. These better estimates are achievable at the expense of assumptions (PU-0)-(PU-3) on f and $\mathbf{X}$.

Assumption (PU-1) means that $\pi_{\mathbf{x}} = \prod_{i=1}^{d_\mathrm{s}} \pi_{\mathrm{x}_i}$ and consequently $L^2_{\pi_{\mathbf{x}}}(S) = \prod_{i=1}^{d_\mathrm{s}} L^2_{\pi_{\mathrm{x}_i}}(S_i)$. This implies that we can construct a basis for $L^2_{\pi_{\mathbf{x}}}(S)$ in terms of the tensor product of basis for $L^2_{\pi_{\mathrm{x}_i}}(S_i)$. Thus for now we will focus on the theory regarding $L^2_{\pi_{\mathrm{x}}}(S)$ space with one dimensional support.

For the sake of simplicity, backed by assumption (PU-1) and noticing that $\mathscr{B}(S) \subset \mathscr{B}(\mathbb{R})$, we will extend the domain of the measure $\pi_{\mathrm{x}} : \mathscr{B}(S) \to [0,1]$ to $\tilde{\pi}_{\mathrm{x}} : \mathscr{B}(\mathbb{R}) \to [0,1]$ assigning measure zero to the sets $A \setminus S$ for $A \in \mathscr{B}(\mathbb{R})$, i.e.

$$\tilde{\pi}_{\mathrm{x}}(A) = \pi_{\mathrm{x}}(A \setminus S^c) \quad \text{for } A \in \mathscr{B}(\mathbb{R}) \,. \tag{5.4}$$

In order to reduce the notation we will denote $\tilde{\pi}_{\mathrm{x}}$ by π_{x}.

Let $\{\phi_j\}_{j=0}^{\infty}$ be an orthonormal basis for $L^2_{\pi_{\mathrm{x}}}(\mathbb{R})$ – its construction will be discussed later. Then $(\phi_i, \phi_j)_{L^2_{\pi_{\mathrm{x}}}(\mathbb{R})} = \delta_{ij}$ and for any $f \in L^2_{\pi_{\mathrm{x}}}(\mathbb{R})$ – assumption (PU-2) – we have that

$$f = \sum_{j=0}^{\infty} \hat{f}_j \phi_j \qquad \hat{f}_j = (f, \phi_j)_{L^2_{\pi_{\mathrm{x}}}(\mathbb{R})} \,. \tag{5.5}$$

For $N \geq 0$, we define the *projection* operator $\mathcal{P}_N : L^2_{\pi_{\mathrm{x}}}(\mathbb{R}) \to \operatorname{span}\left(\{\phi_j\}_{j=0}^{N}\right)$ as

$$\mathcal{P}_N f = \sum_{j=0}^{N} \hat{f}_j \ \phi_j \qquad \hat{f}_j = (f, \phi_j)_{L^2_{\pi_{\mathrm{x}}}(\mathbb{R})} \,. \tag{5.6}$$

Orthonormal basis for $L^2_{\pi_{\mathrm{x}}}(\mathbb{R})$, where π_{x} is a probability distribution, are formed by orthonormal polynomials, i.e. $\{\phi_j\}_{j=0}^{\infty}$ are polynomials such that

$$\begin{aligned} &(\phi_i, \phi_j)_{L^2_{\pi_{\mathrm{x}}}(\mathbb{R})} = \delta_{ij} \\ &\int_{\mathbb{R}} \phi_i(x) \pi_{\mathrm{x}}(\,\mathrm{d}x) = \begin{cases} 1 & \text{if } i = 0 \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{5.7}$$

The orthonormal basis $\{\phi_j\}_{j=0}^{\infty}$ is usually sorted with increasing polynomial orders. Orthogonal polynomials are presented in appendix C. The L^2 convergence of (5.6) to $f \in \mathcal{H}^k_{\mu_{\mathrm{x}}}(\mathbb{R})$ – assumption (PU-3) – is given by [26, 28, 33]:

$$\|f - \mathcal{P}_N f\|_{L^2_{\pi_{\mathrm{x}}}(\mathbb{R})} \leq C(k) N^{-k} |f|_{\mathbb{R}, \pi_{\mathrm{x}}, k} \,. \tag{5.8}$$

Thanks to assumption (PU-1), the construction of a projection operator for dimension $d_{\mathrm{s}} > 1$ can be achieved by the tensor product of the basis $\{\phi_{i,j}\}_{j=0}^{\infty}$ for $L^2_{\pi_{\mathrm{x}_i}}(\mathbb{R})$. For the multi-index $\mathbf{j} = (j_1, \ldots, j_{d_{\mathrm{s}}})$, we denote $\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{\infty}$ the basis for $L^2_{\pi_{\mathbf{x}}}(\mathbb{R}^{d_{\mathrm{s}}})$, with $\Phi_{\mathbf{j}} = \phi_{1,j_1} \otimes \ldots \otimes \phi_{d_{\mathrm{s}}, j_{d_{\mathrm{s}}}}$ and total order $|\mathbf{j}| = j_1 + \ldots + j_{d_{\mathrm{s}}}$. For $N \geq 0$ we define the *projection* operator $\mathcal{P}_N : L^2_{\pi_{\mathbf{x}}}(\mathbb{R}^{d_{\mathrm{s}}}) \to \operatorname{span}\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=\mathbf{0}}^{N}\right)$ as

$$\mathcal{P}_N f = \sum_{|\mathbf{j}|=0}^{N} \hat{f}_{\mathbf{j}} \Phi_{\mathbf{j}} \qquad \hat{f}_{\mathbf{j}} = (f, \Phi_{\mathbf{j}})_{L^2_{\pi_{\mathbf{x}}}(\mathbb{R}^{d_{\mathrm{s}}})} \,. \tag{5.9}$$

In practice (5.9) provides a *surrogate* model for f, which lives in the *simplex tensorized space* $\operatorname{span}\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{N}\right)$ with

$$\dim\left(\operatorname{span}\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{N}\right)\right) = \binom{N+d_{\mathrm{s}}}{N} . \tag{5.10}$$

One could also select the *fully tensorized space* $\operatorname{span}\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0=\mathbf{0}}^{N}\right)$, where $|\mathbf{j}|_0 = \max(\mathbf{j})$, with

$$\dim\left(\operatorname{span}\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0=0}^{N}\right)\right) = (N+1)^{d_{\mathrm{s}}} . \tag{5.11}$$

We can see that (5.11) grows faster than (5.10) as N and d_{s} grow. These are not the only choices available and we will see in the following that finding a space with a mild dimensional growth with respect to d_{s} is crucial for the application of PC to high-dimensional problems.

The statistical moments of $f(\mathbf{X})$ can be obtained from the expansion (5.5) and the properties in (5.7):

$$\begin{aligned}
\mathbf{E}\left[f\right]_{\pi_{\mathbf{x}}} &= \mathbf{E}\left[\sum_{|\mathbf{j}|=0}^{\infty} \hat{f}_{\mathbf{j}}\Phi_{\mathbf{j}}\right]_{\pi_{\mathbf{x}}} = \sum_{|\mathbf{j}|=0}^{\infty} \hat{f}_{\mathbf{j}}\mathbf{E}\left[\Phi_{\mathbf{j}}\right]_{\pi_{\mathbf{x}}} = \hat{f}_{\mathbf{0}}\mathbf{E}\left[\Phi_{\mathbf{0}}\right]_{\pi_{\mathbf{x}}} , \\
\mathbf{V}\left[f\right]_{\pi_{\mathbf{x}}} &= \mathbf{E}\left[\left(\left(\sum_{|\mathbf{i}|,|\mathbf{j}|=0}^{\infty} \hat{f}_{\mathbf{i}}\hat{f}_{\mathbf{j}}\Phi_{\mathbf{i}}\Phi_{\mathbf{j}}\right) - \mathbf{E}\left[f\right]_{\pi_{\mathbf{x}}}\right)^2\right]_{\pi_{\mathbf{x}}} = \sum_{|\mathbf{i}|=1}^{\infty} \hat{f}_{\mathbf{i}}^2 ,
\end{aligned} \tag{5.12}$$

at an insignificant additional computational expense. In the same way, the statistical moments can be approximated using $\mathcal{P}_N f$ by properly truncating the sums in (5.12).

The development up to now has relied on the exactness of the inner products in (5.5)-(5.6) and (5.9). In practice one needs to approximate the inner product $\hat{f}_{\mathbf{j}} = (f, \Phi_{\mathbf{j}})_{L^2_{\pi_{\mathbf{x}}}(\mathbb{R}^{d_{\mathrm{s}}})}$ using a discrete inner product based on cubature rules – high-dimensional quadrature rules – and this leads to an error in the projection. Given the distribution π_{x}, the corresponding Gauss-type quadrature rule, defined by the points and weights $\{(x_i, w_i)\}_{i=0}^{N}$ can be constructed as described in appendix C. Then the integral of $g \in L^2_{\mathrm{x}}(\mathbb{R})$ can be approximated using the quadrature rule $\mathcal{Q}_N$:

$$\int_{\mathbb{R}} g(x)\pi_{\mathrm{x}}(\,\mathrm{d}x) \simeq \sum_{i=0}^{N} g(x_i)w_i =: \mathcal{Q}_N g . \tag{5.13}$$

Gauss, Gauss-Radau and Gauss-Lobatto quadrature rules can be applied to open intervals, intervals open on one side and closed intervals respectively, being exact for functions g of polynomial orders up to $2N+1$, $2N$ and $2N-1$

respectively. A basic construction of cubature rules for higher dimension is based on the tensor product of one dimensional quadrature rules:

$$\mathcal{Q}_{\mathbf{N}} = \mathcal{Q}_N \otimes \ldots \otimes \mathcal{Q}_N \ . \tag{5.14}$$

The discrete version of the projection (5.9) is given by the following definition.

Definition 5.1 (Discrete projection) *Let $(\mathbf{z_i}, w_\mathbf{i})_{\mathbf{i}=0}^N$ be a set of quadrature points and weights. The discrete projection of f is defined in terms of the operator $\tilde{\mathcal{P}}_\mathbf{N} : L^2_\pi(\mathbb{R}^{d_s}) \to \text{span}(\{\Phi_\mathbf{i}\}_{\mathbf{i}=0}^\mathbf{N})$, defined by*

$$\tilde{\mathcal{P}}_\mathbf{N} f = \sum_{\mathbf{i}=0}^{\mathbf{N}} \tilde{f}_\mathbf{i} \Phi_\mathbf{i}, \qquad \tilde{f}_\mathbf{i} = \mathcal{Q}_\mathbf{N}(f\Phi_\mathbf{i}) = \sum_{\mathbf{j}=0}^{\mathbf{N}} f(\mathbf{z_j})\Phi_\mathbf{i}(\mathbf{z_j})w_\mathbf{j} \ . \tag{5.15}$$

If the quadrature rule is a Gauss quadrature rule, then the discrete projection will be exact for $f \in \mathbb{P}_\mathbf{N}$, the set of polynomials of degree up to $\mathbf{N}$.

Polynomial chaos methods can be implemented in two different flavors which belong to the class of *Mean Weighted Residual* (MWR) methods [75]: the *Galerkin* method and the *collocation* method. In the following sections we will review both of them.

Aside from the projection operator, we can also define a polynomial interpolation operator. A number of collocation methods can be based on the interpolation operator. Among the author's publications one of these methods appears in [Bigoni et al., 9]. We refer the interested reader to the presentation therein or to more complete books on the topic [36, 37].

A critical condition in order to be able to approximate integrals by cubature rules is that assumption (PU-0) is fulfilled. When this is not the case, then it is not possible to use the desired quadrature points and alternative methods need to employed. To the author's knowledge works related to PC in these cases are still lagging, with the exception of [76–78]. The literature on machine learning proposes alternative approaches based, for example, on reproducing kernel Hilbert space [32], such as the Kriging estimate [79].

Historical note on Polynomial Chaos methods

Polynomial approximation methods for random variables were first introduced by Wiener (1938) [41] as the *Homogeneous chaos*, where any random variable $X \in L^2(\Omega, \mathscr{F}, P)$ was expanded in terms of a truncated series of Hermite polynomials [80] – see appendix C.2 – denoted as *Polynomial Chaos* (PC) expansion. Hermite polynomials form a basis for $L^2_\pi(\mathbb{R})$ with π being the Normal distribution and they turn out to be optimal in representing Gaussian random variables.

The *generalized Polynomial Chaos* (gPC) [46] was introduced as an extension of PC, where polynomials from the Askey-scheme [81], which are orthogonal with respect to different densities, were related to the distributions with the corresponding densities. This allows a faster convergence of the gPC-expansion of random variables with the corresponding distributions. Appendix C lists some of these orthogonal polynomials.

Along the same lines of thought, orthogonal polynomials can be constructed also for arbitrary densities using Gram-Schmidt orthogonalization or by finding their recursion coefficients. A collection of methods to find recursion coefficients for arbitrary densities was implemented by W.Gautschi [33, 82][4].

5.2.1 Galerkin methods

Galerkin methods belong to the class of intrusive methods: this means that an explicit knowledge of the underlying model needs to be available, i.e. (PU-4) must hold. Given the generic dynamical system with random inputs

$$\begin{cases} \mathbf{u}_t = \mathcal{G}(\mathbf{X})\mathbf{u} & (t, \mathbf{x}, \mathbf{X}) \in T \times D \times S \\ \mathcal{B}(\mathbf{X})\mathbf{u} = 0 & (t, \mathbf{x}, \mathbf{X}) \in T \times \partial D \times S \\ \mathbf{u} = \mathbf{u}_0(\mathbf{X}) & (t, \mathbf{x}, \mathbf{X}) \in T = t_0 \times D \times S \end{cases} \tag{4.2}$$

the Galerkin method aims at its spectral discretization with respect to S. In order to obtain a numerical solution of (4.2), T and D need to be discretized as well using one of the methods commonly used for the solution of deterministic PDEs. What is finally obtained is a mixed discretization of the system.

In the Galerkin approach, we first identify the orthonormal basis $\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{\infty}$ for $L^2_{\pi_{\mathbf{x}}}(\mathbb{R}^{d_s})$ and select a truncation parameter $N \geq 0$, defining the orthonormal system $\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{N}$. Disregarding for now the discretization of T and D, every function and operator – linear – in (4.2) is expanded in terms of the orthonormal

[4] A porting to Python is available at https://pypi.python.org/pypi/orthpol/

system $\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{N}$:

$$\begin{aligned}
\mathbf{u}(t,\mathbf{x},\mathbf{X}) &\simeq \mathbf{u}_N(t,\mathbf{x},\mathbf{X}) = \sum_{|\mathbf{j}|=0}^{N} \hat{\mathbf{u}}_{\mathbf{j}}(t,\mathbf{x})\Phi_{\mathbf{j}}(\mathbf{X}) \ , \\
\mathcal{G}(t,\mathbf{x},\mathbf{X}) &\simeq \mathcal{G}_N(t,\mathbf{x},\mathbf{X}) = \sum_{|\mathbf{j}|=0}^{N} \hat{\mathcal{G}}_{\mathbf{j}}(t,\mathbf{x})\Phi_{\mathbf{j}}(\mathbf{X}) \ , \\
\mathcal{B}(t,\mathbf{x},\mathbf{X}) &\simeq \mathcal{B}_N(t,\mathbf{x},\mathbf{X}) = \sum_{|\mathbf{j}|=0}^{N} \hat{\mathcal{B}}_{\mathbf{j}}(t,\mathbf{x})\Phi_{\mathbf{j}}(\mathbf{X}) \ , \\
\mathbf{u}_0(\mathbf{x},\mathbf{X}) &\simeq \mathbf{u}_{0,N}(\mathbf{x},\mathbf{X}) = \sum_{|\mathbf{j}|=0}^{N} \hat{\mathbf{u}}_{0,\mathbf{j}}(\mathbf{x})\Phi_{\mathbf{j}}(\mathbf{X}) \ .
\end{aligned} \tag{5.16}$$

In the following we will call $\{\hat{\mathbf{u}}_{\mathbf{j}}(t,\mathbf{x})\}_{|\mathbf{j}|=0}^{N}$ the *stochastic modes* of $\mathbf{u}_N(t,\mathbf{x},\mathbf{X})$. Then we will require for the error $(\mathbf{u}-\mathbf{u}_N)$ to be orthogonal to span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{N}\right)$. This is achieved by the solution of the following *weak formulation*: "find $\mathbf{u}_N \in$ span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|=0}^{N}\right)$ such that

$$\begin{cases}
\mathbf{E}\left[\partial_t \mathbf{u}_N \Phi_{\mathbf{i}}\right] = \mathbf{E}\left[\mathcal{G}_N \mathbf{u}_N \Phi_{\mathbf{i}}\right] & (t,\mathbf{x},\mathbf{X}) \in T \times D \times \mathbb{R}^{d_s} \\
\mathbf{E}\left[\mathcal{B}_N \mathbf{u}_N \Phi_{\mathbf{i}}\right] = 0 & (t,\mathbf{x},\mathbf{X}) \in T \times \partial D \times \mathbb{R}^{d_s} \\
\mathbf{E}\left[\mathbf{u}_N \Phi_{\mathbf{i}}\right] = \mathbf{E}\left[\mathbf{u}_0 \Phi_{\mathbf{i}}\right] & (t,\mathbf{x},\mathbf{X}) \in T = t_0 \times D \times \mathbb{R}^{d_s}
\end{cases} \tag{5.17}$$

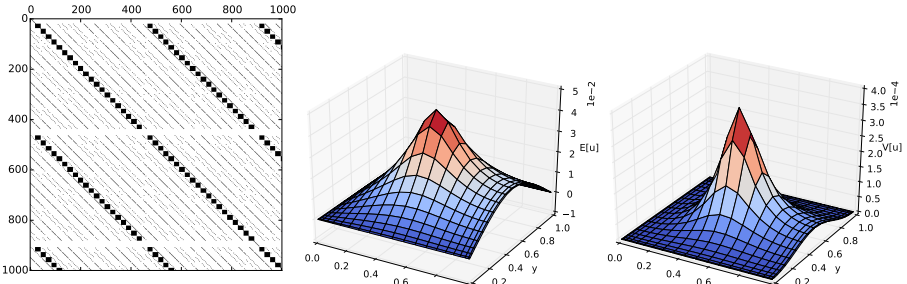
for all $\mathbf{i}$ such that $0 \leq |\mathbf{i}| \leq N$."

If the operator $\mathcal{G}$ or $\mathcal{B}$ is non-linear then the formulation is more involved. Note that when we consider linear deterministic ODEs/PDEs and we rewrite them as stochastic ODEs/PDEs, they can become non-linear ODEs/PDEs due to the dependence of $\mathcal{G}$ on the random input [83]. Then their application to $\mathbf{u}_N \in$ span $\left(\{\Phi_{\mathbf{j}}\}_{\mathbf{j}=0}^{N}\right)$ can easily lead to a result lying outside of span $\left(\{\Phi_{\mathbf{j}}\}_{\mathbf{j}=0}^{N}\right)$. This problem will be discussed in section 5.2.3.

In the following we will use an example in order to explain how the Galerkin method works. We will consider the two dimensional heat equation with the diffusivity coefficient distributed log-normally.

Example 5.1 (Heat Equation) Consider the parametrized heat equation:

$$\begin{cases}
-\nabla \cdot (\kappa(X)\nabla u(\mathbf{x},X)) = h(\mathbf{x}) & \mathbf{x} \in [0,1]^2 \\
u(\mathbf{x},X) = 0 & \mathbf{x} \in \Gamma_D = [0,x_2] \cup [x_1,0] \cup [x_1,1] \\
\frac{\partial_x u}{\partial n}(\mathbf{x},X) = 0 & \mathbf{x} \in \Gamma_N = [1,x_2]
\end{cases} \tag{5.18}$$



(a) Sparsity pattern of a section of the operator

(b) Solution's mean

(c) Solution's variance

Figure 5.5: Example 5.1: Heat equation. Left: sparsity pattern of a part of the discretized operator. Each block of $21^2 \times 21^2$ corresponds to the spatial discretization of the operator. Center and Right: mean and variance of the solution.

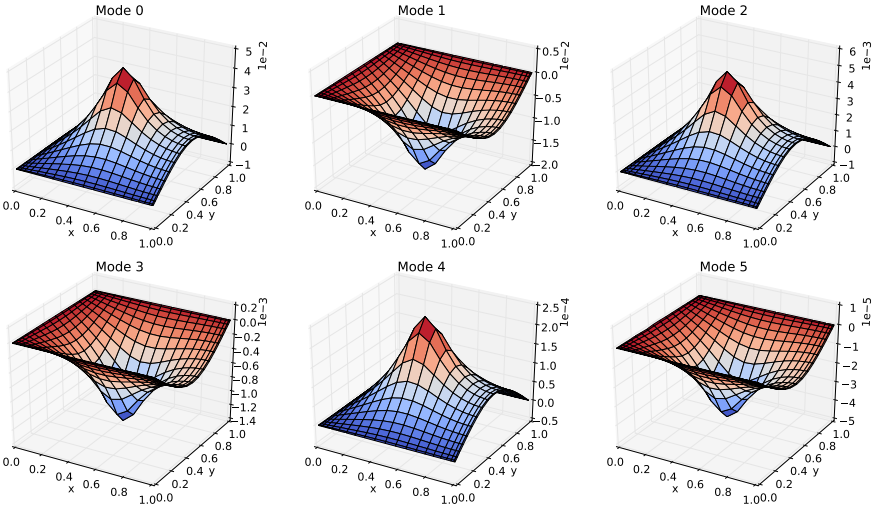


Figure 5.6: Example 5.1: Heat equation. Stochastic modes $\{\hat{u}_i(\mathbf{x})\}_{i=0}^{N}$.

where $h(\mathbf{x}) = \exp\left(-\frac{|\mathbf{x}-\mathbf{x}_0|^2}{l}\right)$, $l = 0.01$, $\kappa = \exp(X)$, $X \sim \mathcal{N}(\mu, \sigma^2)$, $\mu = 0$ and $\sigma = \log(\sqrt{10})/2.85$. Let $\{\phi_i\}_{i=0}^N$ be the normalized Hermite probabilists' polynomials – see appendix C.2.3. These polynomials form a basis for the space of normally distributed random variables. For the interest of the presentation of this example, let us assume that we don't know the analytic relation between κ and X. Then we expand the function u and the diffusivity coefficient κ:

$$\begin{aligned} u(\mathbf{x}, X) &\simeq u_N(\mathbf{x}, X) = \sum_{i=0}^{N} \hat{u}_i(\mathbf{x})\Phi_i(X) \\ \kappa(X) &\simeq \kappa_N(X) = \sum_{i=0}^{N} \hat{\kappa}_i\Phi_i(X) \end{aligned} \tag{5.19}$$

Next we define the weak formulation of problem (5.18): "Find $u_N(\mathbf{x}, X) \in \mathcal{C}^2([0,1]^2) \otimes \text{span}\left(\{\phi_i\}_{i=0}^N\right)$ such that:

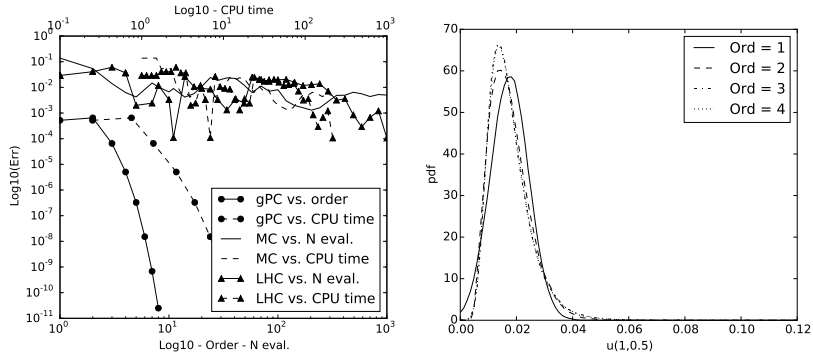
$$\begin{cases} \mathbf{E}\left[-\nabla\left(\kappa_N(X)\nabla u_N(\mathbf{x}, X)\right)\Phi_k\right] = \mathbf{E}\left[h(\mathbf{x})\Phi_k\right] & \mathbf{x} \in [0,1]^2 \\ \mathbf{E}\left[u_N(\mathbf{x}, X)\Phi_k\right] = 0 & \mathbf{x} \in \Gamma_D \\ \mathbf{E}\left[\frac{\partial_x u_N}{\partial n}(\mathbf{x}, X)\Phi_k\right] = 0 & \mathbf{x} \in \Gamma_N \end{cases} \tag{5.20}$$

for all $0 \leq k \leq N$". Using the fact that h is deterministic, and using the properties of orthonormal polynomials, we get:

$$\begin{cases} \sum_{i,j=0}^{N} \hat{\kappa}_j\left(-\nabla^2\hat{u}_i(\mathbf{x})\right)\mathbf{E}\left[\Phi_i\Phi_j\Phi_k\right] = \delta_{0,k}h(\mathbf{x})\mathbf{E}\left[\Phi_k\right] & \mathbf{x} \in [0,1]^2 \\ \hat{u}_k(\mathbf{x}) = 0 & \mathbf{x} \in \Gamma_D \\ \frac{\partial_x\hat{u}_k}{\partial n}(\mathbf{x}) = 0 & \mathbf{x} \in \Gamma_N \end{cases} \tag{5.21}$$

for all $0 \leq k \leq N$. Now each stochastic mode $\{\hat{u}_i(\mathbf{x})\}_{i=0}^N$ can be discretized in space with one of the many discretization schemes available for deterministic PDEs. In this case we chose to use a spectral discretization also in the spatial direction with $N_x = 21$ collocation points for each direction [26, 29], leading to $N_{\mathbf{x}} = 21^2$ d.o.f. for each stochastic mode. Thus, the weak formulation (5.21) results in a sparse linear system of equations with $N \times N_{\mathbf{x}}$ unknowns. The sparsity pattern is shown in figure 5.5a. The solution of such system leads to the stochastic modes shown in figure 5.6. Using these modes and relation (5.12) we can estimate the mean and the variance of the fields as shown in figure 5.5b and 5.5c.

Figure 5.7a compares the convergence rates of the PC Galerkin method with the MC and LHC methods[a]. The convergence rate of the PC Galerkin method is faster than algebraic as the estimate (5.8) predicts for smooth solutions. The approximation of the solution u_N can be used to estimate the PDF of



(a) Convergence rates (b) PDF of $u_N([1, 0.5], X)$

Figure 5.7: Example 5.1: Heat equation. Left: Convergence of the Galerkin method (gPC), MC and LHC. The timings consider both the assembly and the solution of the system for all the methods. The error is computed as $\|\mathbf{E}\left[u_N\right]_{L^2_\pi(\mathbb{R})} - \mathbf{E}\left[u_{\text{ref}}\right]_{L^2_\pi(\mathbb{R})}\|_{L^2(D)}$ where u_{ref} is a high-accuracy reference solution computed with the Galerkin method $N = 10$. Right: convergence of the PDF ρ_N associated with the probability distribution π_N of $u_N([1, 0.5], X)$

the solution. In fact, one can sample from $X \sim \mathcal{N}(\mu, \sigma^2)$ and use (5.19) to retrieve approximate samples of $u(X) \sim \pi_u$. This operation has a negligible computational cost compared to the evaluation of (5.18) for the samples of X, which a MC method would use. The KDE method can then be used to approximate the PDF ρ_u as shown in figure 5.7b.

[a]The timings include both the assembly times and the solution times and have been obtained using a routine written in Python. Better timings could be achieved using a lower lever programming language, but this was not necessary for the sake of this comparison.

5.2.2 Collocation methods

Unlike the Galerkin method, the collocation method is non-intrusive. This means that no knowledge of the underlying model is required, but only the assumptions (PU-0)-(PU-3) need to hold. The collocation methods can be based on projection operators or interpolation operators. We will present here the projection approach, where we use the discrete projection presented in definition 5.1:

$$\tilde{\mathcal{P}}_{\mathbf{N}} f = \sum_{\mathbf{i}=0}^{\mathbf{N}} \tilde{f}_{\mathbf{i}} \Phi_{\mathbf{i}}, \qquad \tilde{f}_{\mathbf{i}} = \mathcal{Q}_{\mathbf{N}}(f\Phi_{\mathbf{i}}) = \sum_{\mathbf{j}=0}^{\mathbf{N}} f(\mathbf{z_j})\Phi_{\mathbf{i}}(\mathbf{z_j})w_{\mathbf{j}} \,. \tag{5.15}$$

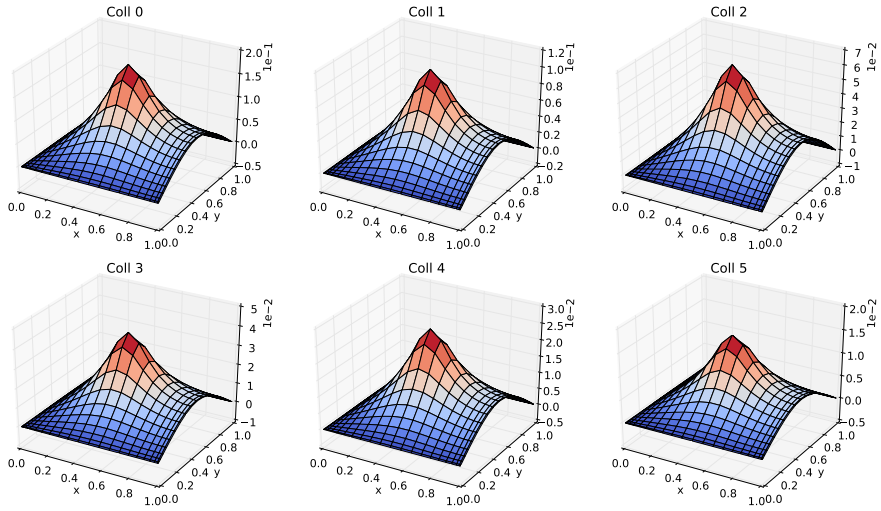


Figure 5.8: Example 5.2: Heat equation. Solutions $\{u(\mathbf{x}, z_i)\}_{i=0}^{5}$ at the $\{z_i\}_{i=0}^{5}$ collocation nodes.

One needs to define a quadrature $\mathcal{Q}_{\mathbf{N}}$ based on point $\{\mathbf{z_j}, w_{\mathbf{j}}\}_{\mathbf{j}=0}^{N}$ and evaluate $\{f(\mathbf{z_j})\}_{\mathbf{j}=0}^{N}$. These values can then be used in (5.15) to find the desired approximation.

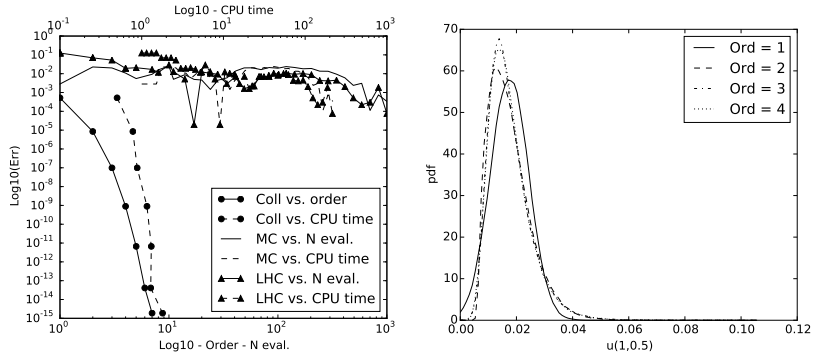
We will use the same example used for the Galerkin method in order to introduce the collocation method.

Example 5.2 (Heat Equation) Consider the parametrized QoI function:

$$\begin{aligned} f : \mathbb{R} &\to \mathcal{C}^2([0,1]^2) \\ X &\mapsto u(\cdot, X) \end{aligned} \tag{5.22}$$

where X and u are as in example 5.1. We construct the quadrature rule $\mathcal{Q}_N$ defined by the points and weights $\{z_i, w_i\}_{i=0}^{N}$ based on the Gaussian distribution and the corresponding Hermite polynomials. Then we evaluate $\{f(z_i)\}_{i=0}^{N}$ solving the deterministic system (5.18). The solutions for $N = 5$ are shown in figure 5.8. These solutions can be used in the quadrature formula (5.15), obtaining the corresponding stochastic modes $\{\tilde{f}_i\}_{i=0}^{N}$. Now the mean and the variance can be approximated by (5.12).

The convergence rate of the collocation method is compared to the MC and the LHC methods in figure 5.7. We can see that the convergence rate achieved is



(a) Convergence rates (b) PDF of $u_N([1, 0.5], X)$

Figure 5.9: Example 5.2: Heat equation. Left: Convergence of the collocation method (gPC), MC and LHC. The timings consider both the assembly and the solution of the system for all the methods. The error is computed as $\|\mathbf{E}\left[u_N\right]_{L^2_\pi(\mathbb{R})} - \mathbf{E}\left[u_{\text{ref}}\right]_{L^2_\pi(\mathbb{R})}\|_{L^2(D)}$ where u_{ref} is a high-accuracy reference solution computed with the Galerkin method $N = 10$. Right: convergence of the PDF ρ_N associated with the probability distribution π_N of $u_N([1, 0.5], X)$

similar to the one obtained by the Galerkin method. The difference in the total accuracy at the machine precision is due to the usage of a Krylov method in the Galerkin case, for which the fixed tolerance is preventing the achievement of the machine precision accuracy. When the convergence rate is related to the CPU timing, we can see that the collocation method outperforms the Galerkin method. The Galerkin problem was preconditioned using an Incomplete LU preconditioner, without doing any additional analysis with regard to other preconditioners. It is thus possible that the results for the Galerkin method could be improved using a more suitable preconditioner.

5.2.3 Limitations of Polynomial Chaos

We can draw few observations already from the example presented for the Galerkin method and the collocation method. A part from requiring the knowledge of the underlying problem – assumption (PU-4) –, the Galerkin method is more cumbersome in its implementation because one needs to construct a mixed discretization of both the stochastic and the deterministic part of the problem. But for the assumptions (PU-0)-(PU-3), the collocation method requires the same knowledge of the model that a random sampling method requires, making it more flexible than the Galerkin approach. This is particularly useful when the deterministic solver that one needs to use is very complex or the source code of

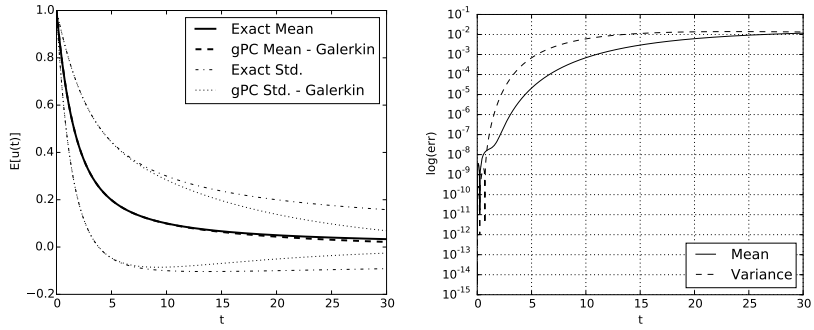
its implementation is not available, as it happens for example with proprietary software. Furthermore, the Galerkin method requires additional tuning in order to solve the big system of discretized equations, which, in spite of being sparse, can quickly become problematic when one considers multiple random inputs. Thus, iterative methods such as Krylov solvers or multigrid solvers [84] need to be used. Ad-hoc preconditioners and algorithms which take into consideration the particular sparsity pattern of the operators can be designed to improve the convergence rate of these methods [85]. However, the complexity of developing these new codes makes Galerkin methods not very attractive in situations where the deterministic problems are very complex and/or big investments have already been made for the development of deterministic solvers and/or the human resources are limited. In these situations, collocation methods are instead very attractive, because they require very little additional implementation with respect to the implementation required for the deterministic solver.

In spite of being cumbersome, Galerkin methods provide a good insight into the problems by giving the user direct control over the stochastic modes, whereas collocation methods can reconstruct the modes only through quadrature or matrix inversion [36, 37].

This control is actually crucial in non-linear problems, where two common phenomena occur: (1) the solution does not belong to the space spanned by the orthonormal system used for the random input, (2) the basis selected is not optimal anymore with respect to the probability measure of the solution. With this perspective, the growth in magnitude of the higher stochastic modes is a good indicator that one of these two problems is occurring. One possible workaround to this problem is the re-orthogonalization of the basis as proposed in [83, 86]. Alternatively one can enrich the orthogonal system by adding orthogonal components, by the construction of Multi-Element gPC (MEgPC) basis functions [87–89] or by the employment of multi-resolution analysis (MRA) [90–92]

In the following example we will present one of these problems and use the re-orthonormalization proposed in [83] to alleviate it. We will work on the linear test equation $\partial_t u = -ku$ and we will consider its stochastic formulation where the decay coefficient is a function of a random input. This will transform the corresponding equation to a non-linear equation due to the quadratic dependence on the random input, as described in [83].

Example 5.3 (Test equation with random input)



(a) Mean and Std. (b) Error

Figure 5.10: Example 5.3: Test equation with random input. Left: Mean and standard deviation Right: Time-dependent errors in mean and variance.

Consider the parametrized test equation

$$\begin{aligned} \partial_t u(t,X) &= -k(X)u(t,X), \qquad u(0,X) = u_0 \,, \\ X \sim \mathcal{U}([-1,1]), \qquad \rho_{\mathrm{x}}(X) &= \frac{1}{2} \qquad k(X) = \frac{1}{2}X + \frac{1}{2} \,. \end{aligned} \tag{5.23}$$

Let $\{\phi_i\}_{i=0}^N$ be normalized Legendre polynomials – see appendix C. Legendre polynomials form a basis for the space of uniformly distributed random variables, and thus are suitable for this case. Let us expand the function u, the linear operator k and the initial condition u_0 in terms of the Legendre polynomials:

$$\begin{aligned} u(t,X) &\simeq u_N(t,X) = \sum_{i=0}^{N} \hat{u}_i(t)\phi_i(X) \,, \\ k(X) &\simeq k_N(X) = \sum_{i=0}^{N} \hat{k}_i\phi_i(X) \,, \\ u_0(X) &\simeq u_{0,N}(X) = \sum_{i=0}^{N} \hat{u}_{0,i}\phi_i(X) \,. \end{aligned} \tag{5.24}$$

Using a quadrature rule or by analytic calculation, one finds that

$$\begin{aligned} \hat{k}_0 &= \frac{1}{2\sqrt{2}} \,, \quad \hat{k}_1 = \frac{1}{2\sqrt{6}} \,, \quad \hat{k}_i = 0 \ \text{ for } i > 1 \,, \\ \hat{u}_{0,0} &= \frac{u_0}{\sqrt{2}} \,, \quad \hat{u}_{0,i} = 0 \ \text{ for } i > 0 \,. \end{aligned} \tag{5.25}$$

Next we define the weak formulation of the problem: "Find $u_N \in \text{span}\left(\{\phi_i\}_{i=0}^N\right)$

such that

$$\begin{aligned}\mathbf{E}\left[\partial_t u_N(t,X)\phi_k(X)\right]_{\rho_\mathrm{x}} &= \mathbf{E}\left[-k(X)u(t,X)\phi_k(X)\right]_{\rho_\mathrm{x}}\\ \partial_t \hat{u}_k(t) &= \sum_{i,j=0}^{N} \hat{k}_i \hat{u}_j(t) \underbrace{\mathbf{E}\left[\phi_i(X)\phi_j(X)\phi_k(X)\right]_{\rho_\mathrm{x}}}_{E_{ijk}}\end{aligned} \tag{5.26}$$

for all $0 \leq k \leq N$." This turns the test equation into a set of N coupled ODEs, where the coupling term E can be precomputed and the initial conditions are given by $\{\hat{u}_{0,i}\}_{i=0}^{N}$ in (5.25). Figure 5.10a shows the time dependent mean and standard deviation obtained using an Adams-Bashforth 4-th order numerical integrator [23, 93] on the system of N ODEs. Note that the variance is not very representative here for the description of the distribution of the solution. In fact we can see that the standard deviation attains also negative values, and thus it extends to a region where no realization of the solution of the system (5.23) would ever be. Indeed the distribution π_u of the solution is characterized by a bigger and bigger skewness as time increases.

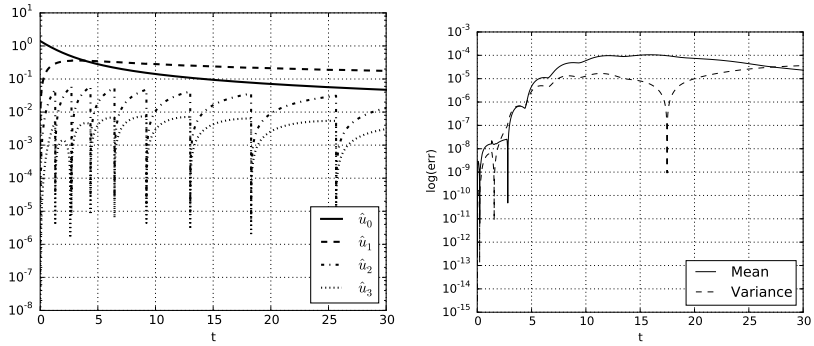
Figure 5.10b shows the magnitude of the error in the estimation of the mean and the variance. We can notice that the approximation degrades over time due to what is known as the stochastic drift. A possible remedy to this effect is a re-orthonormalization of the basis functions with respect to the probability measure of the solution [83]. This is achieved using Gram-Schmidt orthogonalization [94] at every integration time t when the magnitude of one of the stochastic modes $\{u_i(t)\}_{i=2}^{N}$ exceeds $|u_1(t)|/\alpha$, where $\alpha > 1$ determines the frequencies with which the re-orthonormalization should occur.

Figure 5.11a shows the time-dependent magnitudes of the stochastic modes: the basis corresponding to the modes are re-orthonormalized when the magnitude of the stochastic modes $\{u_i(t)\}_{i=2}^{N}$ increase too much. This re-orthonormalization leads to the periodic decay of the highest stochastic modes. In figure 5.11b we see that the error is better bounded for both the mean and the variance of the solution.

All the examples solved with PC up to here have only a one dimensional source of uncertainty. From the Galerkin perspective, assumption (PU-1) allows the construction of multidimensional basis for $L^2_{\pi_\mathbf{x}}(\mathbb{R}^{d_\mathrm{s}})$ as the result of tensor product of one dimensional basis functions for $L^2_{\pi_{\mathrm{x}_i}}(\mathbb{R})$. In (5.11) we already saw that

$$\#\{\Phi_\mathbf{j}\}_{|\mathbf{j}|_0 \leq N} = \dim\left(\operatorname{span}\left(\{\Phi_\mathbf{j}\}_{|\mathbf{j}|_0 \leq N}\right)\right) = (N+1)^{d_\mathrm{s}}\,. \tag{5.27}$$

This means that as d_s increases, one needs to determine an exponentially growing number of stochastic modes. This quickly becomes prohibitive and goes under the name of *curse of dimensionality* [95]. The collocation method suffers



(a) Stochastic modes (b) Error

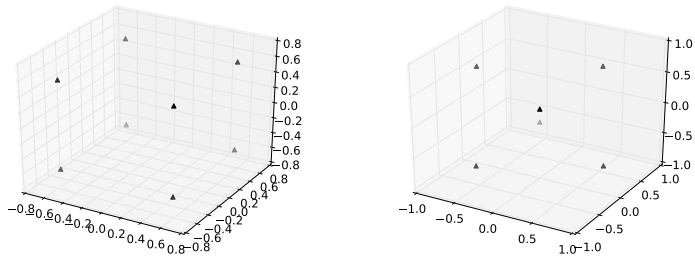
Figure 5.11: Example 5.3: Test equation with random input. Left: Time-dependent magnitude of the first four stochastic modes. Right: Time-dependent errors in mean and variance.

of the same deficiency when high dimensional cubature rules are simply constructed using the full tensor product of one dimensional quadrature rules. This does not come as a surprise because these cubature rules are strictly related to the space span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0 \leq N}\right)$. If on one hand the fully tensorized Galerkin method requires the solution of a system of equations for an exponentially growing number of unknowns, on the other hand the fully tensorized collocation method requires the evaluation of the QoI function on an exponentially growing number of points. In the next section we will present several methods appeared in the last decade aimed to the alleviation of the curse of dimensionality on PC based methods.

5.2.4 Polynomial chaos in high dimensions

The bottleneck in the application of PC in high-dimension is the curse of dimensionality: the dimension of the space spanned by the orthonormal system $\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0 \leq N}$ grows exponentially with the dimension d_{s}, and the computational work required to identify an approximation grows at least as fast.

To the knowledge of the author, all the available techniques for tackling the curse of dimensionality aim at the identification of an optimal subspace of span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0 \leq N}\right)$, where the optimality is meant in terms of the approximation error on the QoI function and in terms of the dimension of the subspace. From this perspective the simplex tensorized space span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}| \leq N}\right)$ with dimension $\binom{N+d_{\mathrm{s}}}{N}$ – c.f. (5.10) – is a good *a priori* candidate subspace of span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0 \leq N}\right)$. However, since it is an *a priori* construction, it does not take into consideration



(a) Full tensor (b) Stroud's rule

Figure 5.12: Left: points belonging to the full tensor product cubature rule of order 3. Right: points belonging to the Stroud's rule of order 3.

the approximation error on the QoI function.

In section 5.2.4.1 we will discuss, without the presumption of completeness, some of the research directions in the field of PC for high-dimensional problems. Section 5.3 will present the High Dimensional Model Representation (HDMR) which will be used in chapter 6 and in some of the practical applications. Section 5.2.4.2 will present the Spectral Tensor-Train decomposition which is a novel technique for the alleviation of the curse of dimensionality.

5.2.4.1 Research directions

When the available computational resources are limited with respect to the size of the problem and only few simulations can be run within an acceptable amount of time, one could wonder on how many collocation points will be needed to obtain an approximation of a fixed total order N with respect to the dimension d_{s}. It turns out that using *Stroud's rules* [96–98] one can construct cubature rules – not based on the tensorization of one dimensional quadrature rules – of order 2 with $d_{\mathrm{s}} + 1$ points. Furthermore, for the symmetric Gaussian and Beta distributions, one can achieve order 3 using only $2d_{\mathrm{s}}$ points. Figure 5.12 shows a comparison between the Gauss cubature rule obtained by the full tensor product of one dimensional rules and the Stroud's rule of the same order of integration[5] 3. A part from the estimation of the first statistics of a random variable, the Stroud's rules find little usage in the construction of PC approximations. In fact, in order to construct these approximations with the projection approach (5.9), one must approximate the inner product $(f, \Phi_{\mathbf{i}})$ by a quadrature rule to obtain the discrete projection (5.15). If the selected rule is a Stroud's rule of order 2 or order 3, the maximum polynomial degree of f must be 1, in order for

[5]Note that the Gauss cubature rule has an accuracy of $2N + 1$.

the approximation to be exact. Thus this approach allows, at its best, only the construction of order 1 PC approximations.

A more flexible collocation approach is given by *Sparse Grids*. This method is based on nested one dimensional quadrature rules such as the Kronrod-Patterson [99, 100], the Clenshaw-Curtis [101, 102] and the Fejèr rules [102, 103]. These rules have the property that doubling the degree of accuracy of the rule – in Sparse Grids terminology this is equivalent to augmenting the level – leads to the usage of points of lower order quadratures. Figure 5.13a shows this property for the Fejèr rule. The nestedness of the rule is useful because one can increase the accuracy of an approximation without wasting already computed values. Gauss-type rules do not possess this property, making it difficult to devise adaptive rules based on them. These nested quadrature rules can be used to construct nested high-dimensional cubature rules through the Smolyak formula [43, 47, 104] which aims at the decomposition of a high-order cubature rule in terms of incremental difference rules over a lower order rule. In practice if we let $\mathcal{Q}_l^{(1)}$ be a level l quadrature rule in one dimension – c.f. (5.13) –, we can define the one dimensional increment formula as

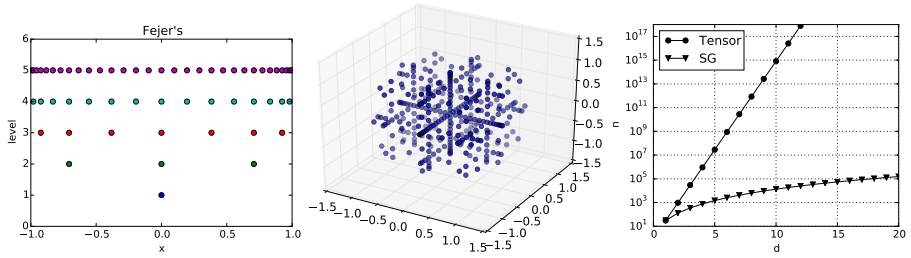
$$\Delta_l := \left(\mathcal{Q}_l^{(1)} - \mathcal{Q}_{l-1}^{(1)}\right) . \tag{5.28}$$

Then the cubature rule of level l in d_s dimensions can be defined in terms of these incremental quadrature rules as:

$$\mathcal{Q}_l^{(d_\mathrm{s})} := \sum_{\mathbf{l}\in L} \left(\Delta_{l_1} \otimes \cdots \otimes \Delta_{l_{d_\mathrm{s}}}\right) . \tag{5.29}$$

In the approach to tackle the curse of dimensionality, the selection of $L = \{\mathbf{l} : |\mathbf{l}| \leq l + N - 1\}$ in place of $L = \{\mathbf{l} : |\mathbf{l}|_0 \leq l\}$ is similar to the selection of the simplex tensorized space span $\left(\{\Phi_\mathbf{j}\}_{|\mathbf{j}|\leq N}\right)$ in place of the bigger fully tensorized space span $\left(\{\Phi_\mathbf{j}\}_{|\mathbf{j}|_0\leq N}\right)$. However this is not the only possible choice. In recent years PC approximations based on sparse grids cubatures, known in the field as *Smolyak pseudo-spectral approximations*, have appeared. The most advanced of these techniques employ an *anisotropic adaptive* approach: in practice the set L is allowed to grow by the addition of multi-indices $\mathbf{l}$ which improve the approximation, and this can end up refining the space more in certain directions than others, i.e. anisotropically. The only constraint for the set L is that it must be admissible with respect to a certain definition of admissibility [48] that allows the Smolyak rule to hold. In order to detect which multi-index needs to be added to L *a posteriori* error estimators have been proposed in [48, 50, 51]. When knowledge of the underlying system is available an *a priori* anisotropy index can be pre-computed as in [49] and used to prioritize the refining directions.

An attractive alternative to Sparse Grids is provided by the methods based on *compressive sensing* [105]. The PC interpolation problem can be phrased as



(a) Fejèr points (b) Fejèr Sparse Grids (c) Fejèr scaling

Figure 5.13: Sparse grids. Left: Points of the Fejèr one dimensional quadrature rule for increasing levels. Center: Points of the sparse grid cubature rule using the Smolyak construction with $L = \{\mathbf{l} : |\mathbf{l}| \leq l + N - 1\}$ on the Fejèr quadrature rule. Right: Scaling of the number of points needed from the Fejèr sparse grid rule with respect to the dimension d_s.

finding $\mathbf{c} \in \mathbb{R}^M$ such that

$$\mathbf{A}\mathbf{c} = \mathbf{f} \,, \tag{5.30}$$

where $\mathbf{A}$ is a generalized Vandermonde matrix of dimensions $n \times M$ based on some orthogonal system with dimension M and $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))^T$. The methods based on compressive sensing recast this PC interpolation problem to a problem of sparse signal recovering. The underlying assumption made in these methods is that only few elements of the candidate orthogonal system $\{\Phi_\mathbf{j}\}_{|\mathbf{j}| \leq N}$ – or the bigger $\{\Phi_\mathbf{j}\}_{|\mathbf{j}|_0 \leq N}$ – need to be used in order to approximate the function. Since the problem associated to ℓ_0 minimization is NP-hard, these methods aim at the solution of the ℓ_1 minimization problem

$$\min \|\mathbf{c}\|_1 \qquad \text{subject to } \|\mathbf{A}\mathbf{c} - \mathbf{f}\|_2 \leq \varepsilon \,, \tag{5.31}$$

which, under some mild conditions, leads to the same result which would be obtained by ℓ_0 minimization and to the identification of a small subspace of span $\left(\{\Phi_\mathbf{j}\}_{|\mathbf{j}| \leq N}\right)$ with orthogonal components corresponding to the multi-indices $\mathbf{j}$ for which $\mathbf{c}_\mathbf{j} \neq 0$. The interested reader is referred to [106–108] for more details regarding these methods.

Galerkin methods for time-dependent problems in high-dimensions can benefit from the direct availability of the stochastic modes. Also in this case the subspace of span $\left(\{\Phi_\mathbf{j}\}_{|\mathbf{j}|_0 \leq N}\right)$ which provides the best approximation is seeked. Once this is identified, only the relevant stochastic modes are evolved in time, reducing the total dimensionality of the problem. All these methods include a strategy for tracking whether the space needs to be enriched with new orthogonal components and thus new stochastic modes needs to be evolved. In the literature these methods go under the name of *adaptive* PC [109] and gPC [110, 111].

In recent years the development of Galerkin methods has focused on the decoupling of the deterministic and stochastic part of differential equations with random input data. These methods use techniques originally developed for the model reduction of experimental measurements of fluid flows, which go under the name of Proper Orthogonal Decomposition (POD) [112, 113]. Subsequently the same ideas have been applied to the solution of deterministic PDEs of reduced dimension under the name of Proper Generalized Decomposition (PGD) [114]. While in deterministic PDEs the POD and the PGD were employed to decouple the temporal and the spatial parts of the problem, in the stochastic setting they are used to decouple the deterministic part from the stochastic part. These techniques go under the name of PGD [115–117] when applied to time-independent problems, and Dynamically Orthogonal (DO) decomposition [118, 119], when applied to time-dependent problems. The advantages carried by these techniques are two-fold: (1) the decoupling allows the use of existing solvers for the deterministic part, (2) the dimensionality of the problem is split, improving the solution performances.

In the following section the spectral tensor-train decomposition [Bigoni et al., 9, 11] will be introduced. This technique is based on linear algebra techniques for the decomposition of tensors, but it also shares much of its theory with PC and POD/PGD techniques.

5.2.4.2 Spectral tensor-train decomposition

The *spectral tensor-train decomposition* (STT-decomposition) is a low-rank decomposition for the spectral approximation of functionals. The goal of this method is the construction of an approximation $\tilde{f}$ of the QoI function f which converges exponentially to f without suffering the curse of dimensionality. In the same fashion of the methods presented in the preceding sections, this method fully exploits assumptions made on f in order to tackle the curse of dimensionality. Here we present its collocation version for the approximation of functionals. Its Galerkin counterpart could be treated similarly, and its application to elliptic PDEs with random inputs is presented in [120, 121].

In the following we collect and summarize the main results we described in [Bigoni et al., 9]. The interested reader is referred to the original article for a more detailed presentation. The construction of the STT-decomposition will proceed in four steps:

1) extension of the discrete tensor-train (DTT) decomposition to the functional tensor-train (FTT) decomposition,
2) characterization of the convergence of the FTT-decomposition depending on the regularity of f,

3) characterization of the regularity of the FTT-decomposition,
4) application of PC to the FTT-decomposition, obtaining the spectral tensor-train (STT) decomposition.

After the presentation of these results, we will present some of the new research directions, which span mainly three topics:

1) automatic reordering,
2) re-weighted DTT-decomposition,
3) anisotropic adaptivity.

These last three features are subject of ongoing development and are scheduled to be soon included in [Bigoni, 11].

Discrete and functional tensor-train decompositions. The *discrete tensor-train decomposition* (DTT-decomposition) was first introduced by I. Oseledets [122] as a robust low-rank alternative to existing tensor decompositions such as the Canonical decomposition (CANDECOMP) and the Tucker decomposition [123, 124]. The DTT-decomposition is a particular case of the hierarchical Tucker decomposition ($\mathcal{H}$-Tucker) [125], but with a simpler implementation which, however, leads to similar performances [126].

Definition 5.2 (DTT-decomposition) *Let* $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_{d_\mathrm{s}}}$*, with* $\boldsymbol{\mathcal{A}}(i_1, \ldots, i_{d_\mathrm{s}})$ *denoting one of its entries. For* $\varepsilon > 0$*, the DTT-decomposition of* $\boldsymbol{\mathcal{A}}$ *is*

$$\boldsymbol{\mathcal{A}}_{TT}(i_1, \ldots, i_{d_\mathrm{s}}) = \sum_{\alpha_0, \ldots, \alpha_{d_\mathrm{s}}=1}^{\mathbf{r}} G_1(\alpha_0, i_1, \alpha_1) \cdots G_{d_\mathrm{s}}(\alpha_{d_\mathrm{s}-1}, i_{d_\mathrm{s}}, \alpha_{d_\mathrm{s}}) \,, \quad (5.32)$$

such that $\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_{TT}\|_F \leq \varepsilon \|\boldsymbol{\mathcal{A}}\|_F$*, where* $\|\cdot\|_F$ *is the Frobenious norm, and* $\langle G_k(\alpha_{k-1}, \cdot, \alpha_k), G_k(\alpha_{k-1}, \cdot, \alpha'_k) \rangle = \delta_{\alpha_k, \alpha'_k} \|G_k(\alpha_{k-1}, \cdot, \alpha_k)\|^2$*, where* $\delta_{\alpha_k, \alpha'_k}$ *is the Kronecker symbol. The vector* $\mathbf{r} = (r_0, \ldots, r_{d_\mathrm{s}})$ *contains the TT-ranks of the decomposition where* $r_0 = r_{d_\mathrm{s}} = 1$*. We will use the notation* $\boldsymbol{\mathcal{E}}_{TT}$ *for the residual tensor* $\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_{TT}$*.*

Such a decomposition is recovered in [122] using the algorithm `TT-SVD`. The main property of the DTT-decomposition is that, given the tensor $\boldsymbol{\mathcal{A}}$, where for simplicity we take $n_1 = \ldots = n_{d_\mathrm{s}} = n$, its DTT-decomposition $\boldsymbol{\mathcal{A}}_{TT}$ requires the storage of only $\mathcal{O}(d_\mathrm{s} n r^2)$ parameters where r is independent of d_s and is optimally selected during the truncation procedure. For more details on the advantages of this decomposition format over the CANDECOMP and the Tucker decompositions we refer to [122] and [Bigoni et al., 9].

In the context of UQ we would consider the tensor $\boldsymbol{\mathcal{A}}$ to be the output of the evaluation of the QoI function f on the tensor grid of points $\boldsymbol{\mathcal{X}} = \times_{j=1}^{d_\mathrm{s}} \mathbf{x}_j$, where $\mathbf{x}_j = (x_{i_j})_{i_j=1}^{n_j}$ for $j = 1, \dots, d_\mathrm{s}$. This is denoted by $\boldsymbol{\mathcal{A}} = f(\boldsymbol{\mathcal{X}})$, with $\boldsymbol{\mathcal{A}}(i_1, \dots, d_\mathrm{s}) = f(x_{i_1}, \dots, x_{i_{d_\mathrm{s}}})$. It makes sense then to define the *functional tensor-train decomposition* (FTT-decomposition) as the functional counterpart of the discrete tensor-train approximation.

Definition 5.3 (FTT-decomposition) *Let (PU-1) and (PU-2) hold for f. For $\mathbf{r} = (1, r_1, \dots, r_{d-1}, 1)$, a TT-rank $\mathbf{r}$ FTT-decomposition of f is:*

$$f_{TT}(\mathbf{x}) := \sum_{\alpha_0, \dots, \alpha_d = 1}^{\mathbf{r}} \gamma_1(\alpha_0, x_1, \alpha_1) \cdots \gamma_d(\alpha_{d-1}, x_d, \alpha_d) \,, \tag{5.33}$$

where $\gamma_i(\alpha_{i-1}, \cdot, \alpha_i) \in L^2_{\pi_i}$ and $\langle \gamma_k(i, \cdot, m), \gamma_k(i, \cdot, n) \rangle_{L^2_{\pi_k}} = \delta_{mn}$. The residual of such approximation will be denoted by $R_{TT} := f - f_{TT}$. We will call $\{\gamma_i\}_{i=1}^d$ the cores of the approximation.

This decomposition is constructed [Bigoni et al., 9] by the recursive application of the functional-SVD.

Definition 5.4 (Functional-SVD) *Let $X \times Y \subset \mathbb{R}^{d_\mathrm{s}}$, $f \in L^2_\pi(X \times Y)$, where $\pi : \mathscr{B}(X \times Y) \to \mathbb{R}$ is a product measure $\pi = \pi_x \times \pi_y$ and let T be the integral operator based on f:*

$$\begin{aligned} T : L^2_{\pi_y}(Y) &\to L^2_{\pi_x}(X) \\ g &\mapsto \int_Y f(x, y) g(y) \pi_y(\,\mathrm{d}y) \end{aligned} \,. \tag{5.34}$$

Let $\{\lambda(i)\}_{i=1}^\infty$ and $\{\psi(x; (i))\}_{i=1}^\infty$, $\{\phi(y; (i))\}_{i=1}^\infty$ be the sets of eigenvalues and eigenfunctions of the integral operators TT^* and T^*T respectively. Then the functional-SVD of f is:

$$f = \sum_{i=1}^\infty \sqrt{\lambda(i)} \psi(\,\cdot\,; (i)) \otimes \phi(\,\cdot\,; (i)) \,. \tag{5.35}$$

The existence, uniqueness and convergence of such decomposition are explained in [Bigoni et al., 9] and are properties of Hilbert-Schmidt kernels [55, 127] such as the function f defined. In different fields the functional-SVD takes different names: this is the SVD in linear algebra [128], the KL-expansion in stochastic processes [59], the POD/PGD for PDEs [113, 114]. This definition has also appeared in [129].

Optimality and convergence of the FTT-decomposition. The first result for the FTT-decomposition is a standard result of optimality inherited from the functional-SVD.

Proposition 5.1 *Let the functional tensor-train decomposition be truncated retaining the largest singular values* $\{\{\sqrt{\lambda_i(\alpha_i)}\}_{\alpha_i=1}^{r_i}\}_{i=1}^{d}$ *of the associated functional-SVDs. Then the approximation* (5.33) *fulfills the condition:*

$$\|R_{TT}\|_{L^2_\pi}^2 = \min_{\substack{g\in L^2_\pi \\ TT\text{-}ranks(g)=\mathbf{r}}} \|f-g\|_{L^2_\pi}^2 = \sum_{i=1}^{d-1}\left(\prod_{j=1}^{i-1} r_j\right)\sum_{\alpha_i=r_i+1}^{\infty}\lambda_i(\alpha_i)\,. \tag{5.36}$$

This means that among all the FTT-decompositions of rank $\mathbf{r}$, the one retaining the biggest singular values is the best in the L^2 sense. However, we need to characterize how the right hand side of (5.36) behaves and we want to link its behavior to the regularity of f. To this end we will use assumption (PU-3). For the sake of simplicity in the following analysis, we will let the ranks be $\mathbf{r} = (r, \dots, r)$.

Theorem 5.1 (FTT-decomposition convergence) *Let* $f \in \mathcal{H}^k_\pi(S)$*, then*

$$\|R_{TT}\|_{L^2_\pi}^2 \le \|f\|^2_{\mathcal{H}^k_\pi(S)}\zeta(k, r+1)\frac{r^{d_s}-r}{r(r-1)} \qquad \text{for } r>1\,, \tag{5.37}$$

where ζ *is the Hurwitz Zeta function. Furthermore*

$$\lim_{r\to\infty}\|R_{TT}\|_{L^2_\pi}^2 \le \|f\|^2_{\mathcal{H}^k_\pi(S)}\frac{1}{(k-1)} \qquad \text{for } k = d_s - 1 \tag{5.38}$$

and

$$\lim_{r\to\infty}\|R_{TT}\|_{L^2_\pi}^2 = 0 \qquad \text{for } k > d_s - 1\,. \tag{5.39}$$

This means that if $f \in \mathcal{H}^k_\pi(S)$, for $k > d_s - 1$, then $f_{TT} \xrightarrow{L^2} f$. See [Bigoni et al., 9] for a detailed proof.

Regularity of the FTT-decomposition. In theorem 5.1 we found a relation between the regularity of f and the convergence of the FTT-decomposition. The goal of this work is going to be the construction of a polynomial approximation of f which complexity scales only linearly with the dimensionality thanks to the FTT-decomposition. To this end one should wonder whether the FTT-decompostion retains any of the regularity properties of f. In particular we will show that the k-th core $\{\gamma_k(\alpha_{k-1}, \cdot, \alpha_k)\}_{\alpha_{k-1},\alpha_k=1}^{r_{k-1},r_k}$ of the FTT-decomposition retains the regularity of f in the k-th direction.

Theorem 5.2 (FTT-decomposition and Sobolev spaces) *Let $S_1 \times \cdots \times S_d = S \subset \mathbb{R}^d$ be closed and bounded, and $f \in L^2_\pi(S)$ be a Hölder continuous function with exponent $\alpha > 1/2$ such that $f \in \mathcal{H}^k_\pi(S)$. Then the FTT-decomposition* (5.33) *is such that $\gamma_j(\alpha_{j-1}, \cdot, \alpha_j) \in \mathcal{H}^k_{\pi_j}(S_j)$ for all j, α_{j-1} and α_j.*

The results above have the limitation of holding for functions defined on closed and bounded domains. In many practical cases encountered in UQ, however, functions are defined on the real line, equipped with a finite measure. As proven in [Bigoni et al., 9], the theorem 5.2 uses a result by Smithies [130, Thm. 14] which hinges on a result by Hardy and Littlewood [131, Thm. 10] on the convergence of Fourier series. This is the only passage in the proof where the closedness and boundedness of the domain is explicitly used. A similar result for an orthogonal system in $L^2_\pi(-\infty, \infty)$, where π is a finite measure, would be sufficient to extend Smithies' result to the real line. To the author's knowledge, the corresponding result for such cases has not been proved in literature.

Since the goal of this section is the construction of an approximation method based on PC, all the results have been obtained with respect to the weak derivatives of f. One may also wonder about the strong regularity of the FTT-decomposition, i.e. with respect to the strong derivatives of f. First we mention a result on the continuity of the FTT-decomposition which follows directly from Mercer's theorem [132].

Proposition 5.2 (Continuity) *Let $S_1 \times \cdots \times S_{d_\mathrm{s}} = S \subset \mathbb{R}^{d_\mathrm{s}}$, and $f \in L^2_\pi(S)$ be a continuous function with FTT-decomposition* (5.33)*. Then $\gamma_i(\alpha_{i-1}, \cdot, \alpha_i)$ are continuous for every i and α_i.*

For a Lipschitz continuous function f, the preservation of the strong derivatives follows from the uniform convergence of the functional-SVD, which is a result by Hammerstein [129, 133]. The following result is mentioned without proof in [Bigoni et al., 9]. Here we provide also its proof.

Proposition 5.3 (Differentiability) *Let $S_1 \times \ldots \times S_{d_\mathrm{s}} = S \subset \mathbb{R}^{d_\mathrm{s}}$ be closed and bounded, and $f \in L^2_\pi(S)$ be a Lipschitz continuous function such that $\frac{\partial^\beta f}{\partial x_1^{\beta_1} \cdots \partial x_{d_\mathrm{s}}^{\beta_{d_\mathrm{s}}}}$ exists and is continuous on S for $\beta = \sum_{i=1}^{d_\mathrm{s}} \beta_i$. Then the FTT-approximation* (5.33) *is such that $\gamma_k(\alpha_{k-1}, \cdot, \alpha_k) \in \mathcal{C}^{\beta_k}(S_k)$ for all k, α_{k-1} and α_k.*

Proof. Let us first show this for the functional-SVD (5.35) of the Lipschitz continuous function $f \in L^2_\pi(X \times Y)$ for which $\frac{\partial^\beta}{\partial x^{\beta_1} \partial^{\beta_2}} f$ exists and is continuous

on the compact set $X \times Y$. We have that

$$\begin{aligned}\frac{1}{\lambda_i}\left\langle \frac{\partial^{\beta_1}}{\partial x^{\beta_1}} f(x,y), \psi_i(y)\right\rangle_{L^2_{\pi_y}(Y)} &= \frac{1}{\lambda_i}\left\langle \frac{\partial^{\beta_1}}{\partial x^{\beta_1}} \sum_{j=1}^{\infty} \lambda_j \phi_j(x)\psi_j(y), \psi_i(y)\right\rangle_{L^2_{\pi_y}(Y)} \\ = \frac{1}{\lambda_i}\left\langle \sum_{j=1}^{\infty} \lambda_j \frac{\partial^{\beta_1}}{\partial x^{\beta_1}} \phi_j(x)\psi_j(y), \psi_i(y)\right\rangle_{L^2_{\pi_y}(Y)} &= \frac{\partial^{\beta_1}}{\partial x^{\beta_1}}\phi_i(x)\end{aligned} \tag{5.40}$$

where the second equality is given by the uniform convergence of the functional-SVD (5.35). Since $\frac{\partial^{\beta_1}}{\partial x^{\beta_1}} f(x,y)$ is continuous by assumption, $\{\psi_i\}_{i=1}^{\infty}$ are continuous by Proposition 5.2, the left hand side of (5.40) is continuous. Thus $\{\phi_i\}_{i=1}^{\infty} \in \mathcal{C}^{\beta_1}(X)$ and $\{\psi_i\}_{i=1}^{\infty} \in \mathcal{C}^{\beta_2}(Y)$.

Since the FTT-decomposition (5.33) is constructed by repeated functional SVDs, then for fixed $\boldsymbol{\alpha} = (\alpha_0, \ldots, \alpha_{d_s})$:

$$\begin{aligned}\frac{\partial^{\beta_k}}{\partial x_k^{\beta_k}}\gamma_k(\alpha_{k-1}, x_k, \alpha_k) = &\frac{1}{\sigma(\boldsymbol{\alpha})}\langle \frac{\partial^{\beta_k}}{\partial x_k^{\beta_k}} f(\mathbf{x}), \gamma_1(\alpha_0, x_1, \alpha_1)\cdots \\ &\gamma_{k-1}(\alpha_{k-2}, x_{k-1}, \alpha_{k-1})\gamma_{k+1}(\alpha_k, x_{k+1}, \alpha_{k+1})\cdots \\ &\gamma_{d_s}(\alpha_{d_s-1}, x_{d_s}, \alpha_{d_s})\rangle\end{aligned} \tag{5.41}$$

exists and is continuous. This means that $\gamma_k(\alpha_{k-1}, \cdot, \alpha_k) \in \mathcal{C}^{\beta_k}(S_k)$ for any k, α_{k-1} and α_k. □

The spectral tensor-train decomposition. We can now apply the PC machinery to the FTT-decomposition. Let $\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0=0}^{N}$ be the fully tensorized orthogonal/orthonormal system with respect to π, where $\Phi_{\mathbf{j}} = \phi_{1,j_1} \otimes \ldots \otimes \phi_{d_s,j_{d_s}}$. The projection operator $\mathcal{P}_N : L^2_\pi(\mathbb{R}^{d_s}) \to \text{span}\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0=0}^{N}\right)$ defined by (5.9) is built upon the tensor product of one dimensional projections $\mathcal{P}_N^{(n)} : L^2_{\pi_n}(\mathbb{R}) \to \text{span}\left(\{\phi_{n,j_n}\}_{j_n=0}^{N}\right)$:

$$\mathcal{P}_N = \mathcal{P}_N^{(1)} \otimes \ldots \otimes \mathcal{P}_N^{(d_s)} \,. \tag{5.42}$$

Then the projection of the FTT-decomposition f_{TT} onto span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0=0}^{N}\right)$ is

$$\begin{aligned}\mathcal{P}_N f_{TT} &= \sum_{\mathbf{i}=0}^{\mathbf{N}} \tilde{c}_{\mathbf{i}} \Phi_{\mathbf{i}} \ , \\ \tilde{c}_{\mathbf{i}} &= \int_{\mathbb{R}^{d_\mathrm{s}}} f_{TT} \Phi_{\mathbf{i}} \pi(\,\mathrm{d}\mathbf{x}) \\ &= \sum_{\alpha_0,\ldots,\alpha_{d_\mathrm{s}}=1}^{\mathbf{r}} \sigma(\boldsymbol{\alpha}) \beta_1(\alpha_0, i_1, \alpha_1) \cdots \beta_{d_\mathrm{s}}(\alpha_{d_\mathrm{s}-1}, i_{d_\mathrm{s}}, \alpha_{d_\mathrm{s}}) \ ,\end{aligned} \tag{5.43}$$

where

$$\begin{aligned}\beta_n(\alpha_{n-1}, i_n, \alpha_n) &= \mathcal{P}_N^{(n)} \gamma_n(\alpha_{n-1}, \cdot, \alpha_n) \\ &= \int_{\mathbb{R}} \gamma_n(\alpha_{n-1}, x_n, \alpha_n) \phi_{n,i_n}(x_n) \pi_n(\,\mathrm{d}x_n) \ .\end{aligned} \tag{5.44}$$

Note that the particular format of the FTT-decomposition allows the usage of one dimensional projection operators $\mathcal{P}_N^{(n)}$ on the cores instead of the d_s-dimensional projection operator $\mathcal{P}_N$ on f. The approximation $\mathcal{P}_N f_{TT}$ is called the *spectral tensor-train decomposition*. Note that the STT-decomposition provides also the tensor of generalized Fourier coefficients $\boldsymbol{\mathcal{C}}_{TT}$ with entries $\boldsymbol{\mathcal{C}}_{TT}(\mathbf{i}) = c_{\mathbf{i}}$, which is already in the format of a DTT-decomposition.

Thanks to theorem 5.1 and standard results on the convergence of polynomial projection approximations [26, 28] we have the following convergence result.

Proposition 5.4 (Convergence of the STT-decomposition) *For $k > d_\mathrm{s} - 1$, let $f \in \mathcal{H}_\pi^k(\mathbb{R}^{d_\mathrm{s}})$, then:*

$$\begin{aligned}\|f - \mathcal{P}_N f_{TT}\|_{L^2_\pi(\mathbb{R}^{d_\mathrm{s}})} \leq & \|f\|_{\mathcal{H}^k_\pi(\mathbb{R}^{d_\mathrm{s}})} \sqrt{\frac{(r+1)^{-(k-1)}}{k-1} \frac{r^{d_\mathrm{s}-1}-1}{r-1}} + \\ & + C(k) N^{-k} |f_{TT}|_{\mathbb{R}^{d_\mathrm{s}},\pi,k} \ .\end{aligned} \tag{5.45}$$

The convergence rate of the STT-decomposition is thus the consequence of the combined benefits of the FTT-decomposition and the PC high-order methods.

Analogous construction and convergence results can be obtained using interpolation operators. We refer the reader to [Bigoni et al., 9] for more details about them.

Practical computation of the STT-decomposition. The construction of the STT-decomposition up to now has relied on exact inner products for the

computation of the projection (5.43). In practice the inner products must be approximated by discrete inner products using the quadratures (5.13)-(5.14). These quadratures define the tensor grid of points $\boldsymbol{\mathcal{X}} = \times_{j=1}^{d_s} \mathbf{x}_j$ and the tensor of weights $\boldsymbol{\mathcal{W}} = \mathbf{w}_1 \circ \cdots \circ \mathbf{w}_{d_s}$, where $\circ$ denotes here the outer product of vectors. The evaluation of f on all the elements of $\boldsymbol{\mathcal{X}}$ provides the tensor $\boldsymbol{\mathcal{A}} = f(\boldsymbol{\mathcal{X}})$ which can be approximated by $\boldsymbol{\mathcal{A}}_{TT}$ using the `TT-SVD` [122].

In spite of providing a decomposition with $\mathcal{O}(d_s n r^2)$ parameters, the `TT-SVD` requires the full computation of $\boldsymbol{\mathcal{A}}$, and thus does not alleviate the curse of dimensionality. Thus, instead of using the `TT-SVD`, we employ the `TT-dmrg-cross` [134, 135] which is a method for the construction of the DTT-decompositions requiring only a sparse sampling of $\boldsymbol{\mathcal{A}}$ and leading to $\mathcal{O}(d_s n r^2)$ function evaluations. The tensor of generalized Fourier coefficients $\boldsymbol{\mathcal{C}}_{TT}$ is then obtained from $\boldsymbol{\mathcal{A}}_{TT}$. This step and the subsequent evaluation of $\mathcal{P}_N f_{TT}$ are detailed in [Bigoni et al., 9]. Note that in both of these steps the algorithms always handle only $\mathcal{O}(d_s n r^2)$ parameters, in contrast to the $\mathcal{O}(n^{d_s})$ parameters required for the computation of $\mathcal{P}_N f$.

The construction of $\boldsymbol{\mathcal{A}}_{TT}$ is further improved by the adoption of the *Quantics tensor-train decomposition* (QTT-decomposition) [120, 121, 136, 137]. This mainly consists in a folding of the tensor $\boldsymbol{\mathcal{A}}$ which leads to better scaling of the number of function evaluations and stored parameters for its DTT-decomposition. If we let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_{d_s}}$, where for simplicity $n_i = \ldots = n_{d_s} = n$ and n is a power of q, i.e. $n = q^m$, then we can reshape $\boldsymbol{\mathcal{A}}$ into the *quantics-tensor* $\boldsymbol{\mathcal{A}}^{(q)} \in \mathbb{R}^{q \times \cdots \times q}$ which is a $(m d_s)$ dimensional tensor. Thus, its approximation $\boldsymbol{\mathcal{A}}_{TT}^{(q)}$ involves $\mathcal{O}(m d_s q r^2)$ parameters. Since the best scaling is obtained when $q = 2$, the practical implementation [Bigoni, 11] of the QTT-decomposition uses a cost free padding technique to obtain $n = 2^m$.

Strengths of the STT-decomposition. The scalability of the STT-decomposition with respect to d_s is characterized by $\mathcal{O}(m d_s r^2)$, where $m = \lceil \log_2 n \rceil$. This property, along with the convergence rate obtained in proposition 5.4 is now checked on two test functions. Other similar results and comparisons with other methods are presented in [Bigoni et al., 9].

Example 5.4 (Convergence rate) Here we will use the modified Genz functions [Bigoni et al., 9], [138, 139] to test the convergence rate of the STT-

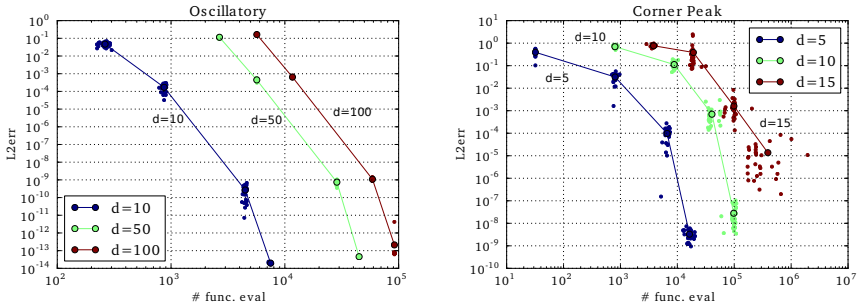


Figure 5.14: Convergence of the STT-decomposition on the modified test functions. For exponentially increasing polynomial order ($2^i - 1$ for $i = 1, \ldots, 4$) and for different dimensions, 30 modified Genz functions have been constructed and approximated by the STT-decomposition. The scattered dots show the L^2 error and the number of function evaluations needed for each of these realizations. The circled dots represent the mean L^2 error and mean number of function evaluations for increasing polynomial order.

decomposition. Let the two QoI functions be

$$\begin{aligned} \text{oscillatory} : f_1(\mathbf{x}) &= \cos\left(2\pi w_1 + \sum_{i=1}^{d_\mathrm{s}} c_i x_i\right) , \\ \text{corner peak} : f_2(\mathbf{x}) &= \left(1 + \sum_{i=1}^{d_\mathrm{s}} c_i x_i\right)^{-(d_\mathrm{s}+1)} , \end{aligned} \tag{5.46}$$

where the parameters w_1 and $\mathbf{c}$ are drawn uniformly from $[0, 1]$. A set of 30 functions are generated for each QoI function and then approximated by STT-decompositions of increasing polynomial order. The L^2 error

$$\frac{\|f - \mathcal{L} f_{TT}\|_{L^2_\pi(\mathbb{R}^{d_\mathrm{s}})}}{\|f\|_{L^2_\pi(\mathbb{R}^{d_\mathrm{s}})}} = \sqrt{\frac{\int_{\mathbb{R}^{d_\mathrm{s}}} (f - \mathcal{L} f_{TT})^2 \pi(\,\mathrm{d}\mathbf{x})}{\int_{\mathbb{R}^{d_\mathrm{s}}} f^2 \pi(\,\mathrm{d}\mathbf{x})}} \tag{5.47}$$

of these approximations are shown in figure 5.14. We can see that the approximation of the "oscillatory" function is considerably easier respect to the approximation of the "corner peak" function. This is due to the fact that the "oscillatory" function has an exact rank-two representation – one can see it using basic trigonometric rules – and thus the $r = 2$ truncation of the FTT-decomposition (5.33) leads to an exact representation of f_1. On the contrary

the "corner peak" function has no exact low-rank representation leading to truncation errors in the FTT-decomposition. In order to limit these errors the ranks $\mathbf{r}$ needs to be increased and this leads to an increased number of function evaluations.

Next, we will show two additional properties which are peculiar of the method. We will show these properties through two examples.

Example 5.5 (Detection of features) This property is actually a property of the `TT-dmrg-cross` algorithm when applied on a tensor or on a quantics-tensor. Let the QoI function be

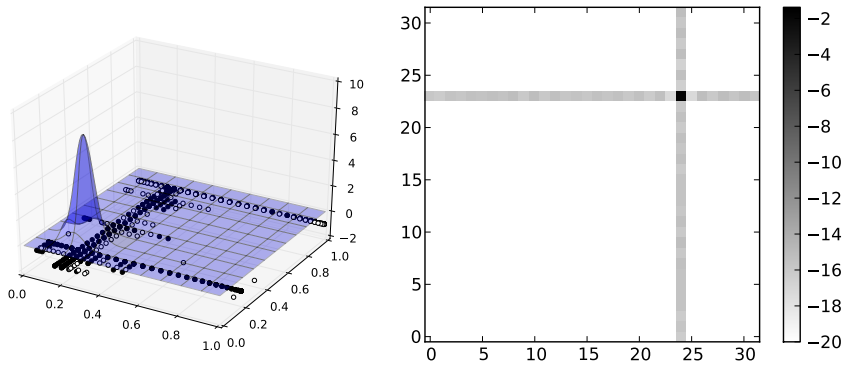
$$f(\mathbf{x}) = \exp\left(-\frac{|\mathbf{x}-\mathbf{x}_0|^2}{2l^2}\right), \tag{5.48}$$

for $\mathbf{x} \in S \equiv [0,1]^{d_s}$, $d_s = 2$, $\mathbf{x}_0 = [0.2, 0.2]$ and $l = 0.05$. This function is smooth and has the exact rank-one representation

$$f(\mathbf{x}) = \exp\left(-\frac{(\mathbf{x}_1-\mathbf{x}_{0,1})^2}{2l^2}\right) \cdot \exp\left(-\frac{(\mathbf{x}_2-\mathbf{x}_{0,2})^2}{2l^2}\right). \tag{5.49}$$

This makes the function relatively easy to be represented by an FTT-decomposition. However, the function is characterized by an off-centered feature which requires high-order quadrature rules in order to be detected. Figure 5.15a shows the function f along with the evaluation points used by the `TT-dmrg-cross` algorithm for the construction of an approximation of accuracy $\varepsilon = 10^{-8}$. Gauss quadrature rules of order 67, i.e. with $N = 33$ points, are constructed for each dimension, determining the candidate points for the evaluation of f. Out of these $N^2 = 1089$ points, the `TT-dmrg-cross` algorithm selects a relatively small subset of 294 points where to evaluate the function. These points are the white and black points in figure 5.15a. The black points are the ones retained for the final approximation and we can see that they have clustered around the function's peak feature at $\mathbf{x}_0$. Examples in higher dimensions are reported in [Bigoni et al., 9].

Example 5.6 (*A posteriori* basis selection) Unlike many other methods which aim at the alleviation of the curse of dimensionality, the STT-decomposition does not make any attempt to reduce the dimension of span $\left(\{\Phi_{\mathbf{j}}\}_{|\mathbf{j}|_0=0}^{N}\right)$. In principle the whole space is used. Here we construct an ad-hoc function to



(a) Features detection (b) *A posteriori* basis selection

Figure 5.15: Left: Features detection example 5.5. The STT-decomposition of the function (5.48) is constructed through the evaluation at the points shown in the figure. The black points are the ones retained by the `TT-dmrg-cross` algorithm for the final approximation. Right: *A posteriori* basis selection example 5.6. The $\log_{10}$ of the magnitude of the generalized Fourier coefficients is shown.

pinpoint this property. Let the QoI function be

$$f(\mathbf{x}) = \prod_{k=1}^{c} \phi_{l_k}(x_{j_k}) \,. \tag{5.50}$$

where $\mathbf{x} \in S \equiv [-1,1]^{d_\mathrm{s}}$, $d_\mathrm{s} = 2$, ϕ_{l_k} are polynomials of order l_k and $l = [23, 24]$. Let us consider the space of polynomials span $\left(\{\Phi_\mathbf{j}\}_{|\mathbf{j}|_0=0}^{N}\right)$ with $N = 31$ and construct the STT-decomposition $\mathcal{P}_N f_{TT}$. Figure 5.15b shows the magnitude of the generalized Fourier coefficients $\boldsymbol{\mathcal{C}}$ of the STT-decomposition in the $\log_{10}$ scale. We can see that the right polynomial order is identified. In spite of the high polynomial order of the function f is reconstructed using only 209 function evaluations out of 1024 candidate points. The following table shows the performances of the STT-decomposition for increasing dimensions and polynomial order $l = [24, \ldots, 24]$:

d_s	Rank r	#f.eval.	#Cand.Points
2	1	209	1024
3	1	626	32768
4	1	1210	1048576
5	1	1442	33554432

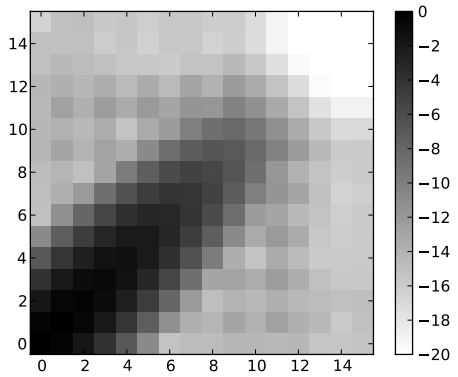


Figure 5.16: Ordering problem example 5.7. Magnitude of the slowly decaying generalized Fourier coefficients of (5.52) in the $\log_{10}$ scale.

The ordering problem. The major weakness of the STT-decomposition as well as of the DTT-decomposition is the ordering problem. Even if this problem does not produce erroneous results, it can lead to a higher number of function evaluations than needed. In the following we will say that a function f has an *analytical low-rank* representation if there exist $\left\{\{g_{i,j}\}_{j=1}^{d_\mathrm{s}}\right\}_{i=1}^{r}$ such that

$$f(\mathbf{x}) \equiv \sum_{i=1}^{r}\prod_{j=1}^{d_\mathrm{s}} g_{i,j}(x_j)\,, \quad \text{for } r<\infty\,. \tag{5.51}$$

The ordering problem will be presented through an example.

Example 5.7 (The ordering problem) Let $S \equiv [-1,1]^{d_\mathrm{s}}$ and consider the sub-cube $S_{j_1}\times\cdots\times S_{j_c}$, where $J=\{j_i\}_{i=1}^{c}\subseteq[1,\ldots,d_\mathrm{s}]$. For $\mathbf{x}\in S$, let the QoI function be

$$f(\mathbf{x}) = \sum_{i_{j_1}=0}^{n_{j_1}}\cdots\sum_{i_{j_c}=0}^{n_{j_c}}\left[\exp\left(-\mathbf{i}^T\boldsymbol{\Sigma}\mathbf{i}\right)\prod_{k=1}^{c}\phi_{i_{j_k}}(x_{j_k})\right]\,, \tag{5.52}$$

where $\boldsymbol{\Sigma}$ is a $c\times c$ matrix defining the level of interaction between different dimensions, $\{\phi_{i_{j_k}}\}_{i_{j_k}=1}^{n_{j_k}}$ are chosen to be the normalized Legendre polynomials, $\mathbf{i}=(i_{j_1},\ldots,i_{j_c})^T$ and the ϕ_{l_k} are possibly high order polynomials. To simplify the notation, we will set $n_{j_k}=n$ for all j_k. For $d_\mathrm{s}=2$, $J=[0,1]$ and

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & -0.9 \\ -0.9 & 1 \end{bmatrix}\,,$$

figure 5.16 shows the decay of the $\log_{10}$ magnitude of the generalized Fourier coefficients of f. Retaining the same accuracy target, the following table reports the ranks and the number of function evaluations used for the construction of the STT-decomposition of f for different dimensions and sets J:

d_s	J	$\mathbf{r}$	#f.eval.	#Cand.Points
2	$[0,1]$	$[1,11,1]$	256	256
5	$[1,2]$	$[1,1,11,1,1,1]$	3935	1048576
5	$[0,4]$	$[1,11,11,11,11,1]$	73307	1048576

Note that the functions relative to the last two entries are totally equal but for the ordering of their axis. However the different ordering causes a 20-fold increase in the number of function evaluations!

The causes of the ordering problem are to be searched in the formats of the DTT-decomposition and the FTT-decomposition. Let us consider the DTT-decomposition defined in (5.32):

$$\boldsymbol{\mathcal{A}}_{TT}(i_1,\ldots,i_{d_s}) = \sum_{\alpha_0,\ldots,\alpha_{d_s}=1}^{\mathbf{r}} G_1(\alpha_0,i_1,\alpha_1)\cdots G_{d_s}(\alpha_{d_s-1},i_{d_s},\alpha_{d_s})\,. \tag{5.53}$$

In terms of the traditional matrix SVD [128] $\mathbf{A}=\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$, where $\mathbf{U}$ and $\mathbf{V}$ are two unitary matrices and $\boldsymbol{\Sigma}=\mathrm{diag}(\sigma)$ is the matrix of singular values, the rank of a matrix $\mathbf{A}$ is given by the number of non-zero values in σ – this corresponds to the number of the independent rows/columns of $\mathbf{A}$. The numerical rank r to which we will refer here is the number of retained singular values $\hat{\boldsymbol{\Sigma}} = \mathrm{diag}\left(\{\sigma_i\}_{i=1}^{r}\right)$ in the SVD of a matrix $\mathbf{A}$ in order to have $\|\mathbf{A}-\mathbf{A}_{\mathrm{SVD}}\|_F \leq \varepsilon$, where $\mathbf{A}_{\mathrm{SVD}}=\mathbf{U}\hat{\boldsymbol{\Sigma}}\mathbf{V}^T$.

If we consider the function of example 5.7, we see that it is constant along all the directions but J. A restriction of the function to the two directions in J produces a function $\tilde{f}$ and an associated matrix $\mathbf{A}=\tilde{f}(\mathbf{X})$ with a high numerical rank.

The truncation performed in order to obtain the DTT-decomposition (5.53) is totally equivalent to the truncation used in the traditional matrix SVD. However, while in the two dimensional case of a matrix the two dimensions are directly related to each other by $\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$, in the DTT-decomposition each core G_i is connected only to the two neighboring cores G_{i-1} and G_{i+1} – the name tensor-train derives from this property. Thus, if $J=[i,j]$ for two neighboring indices $i<j$, the numerical TT-rank $\mathbf{r}$ will be high only between the two cores G_i and G_j. On the contrary, if $i<j$ are not neighboring, all the TT-ranks between the cores $G_i, G_{i+1},\ldots,G_j$ will be high. Tensors with high numerical

TT-ranks require the approximation and the storage of a higher number of parameters as it is clear from (5.53), and thus they lead to an increase in the number of function evaluations.

In the following we will propose a method for the approximation of a correct ordering. A first approach to the problem could be to consider the second order sensitivities of the function[6] – see chapter 6. However, this can be misleading, as we will show in the following non-pathological example.

Example 5.8 (TT-ranks vs. Sensitivity Indicies) Let us consider the QoI function

$$f(\mathbf{x}) = 10x_1x_2 + 10x_2x_3 + \frac{1}{x_1 + x_3 + 1}\,, \tag{5.54}$$

where $\mathbf{x} \in S \equiv [0,1]^{d_\mathrm{s}}$ and $d_\mathrm{s} = 3$. Using the ANOVA decomposition of f, we can compute the second order sensitivities $S_{i,j}$ described in (6.1):

(i,j)	$S_{i,j}$
$(1,2)$	0.051226
$(1,3)$	0.000114
$(2,3)$	0.051226

These values are computed to machine precision with high order cubature rules and a 3-rd order cut-HDMR decomposition – which is exact because $d_\mathrm{s} = 3$. These sensitivities suggest correctly that the contribution given to the variance by the combination of the first and third inputs is significantly smaller than the other combinations. This is also intuitively reasonable observing that the first two summands of the function (5.54) determine a bigger gradient respect to the last summand. These sensitivities would suggest $(1,2,3)$ to be a correct ordering of the dimensions. Unfortunately, the first two summands of (5.54) have an analytical rank-one representation, whereas the last summand has no analytical low-rank representation. Thus the ordering $(1,2,3)$ is actually the only wrong possible ordering!

We can see this by the construction of the STT-decomposition for different tolerances ε and for two different orderings:

[6] This approach was also suggested in [140]

Order:	$(1,2,3)$		$(1,3,2)$	
ε	TT-Ranks	#f.eval.	TT-Ranks	#f.eval.
10^{-1}	$[1,2,2,1]$	263	$[1,2,2,1]$	152
10^{-2}	$[1,2,2,1]$	283	$[1,2,2,1]$	276
10^{-3}	$[1,3,3,1]$	294	$[1,3,2,1]$	289
10^{-4}	$[1,4,4,1]$	340	$[1,4,2,1]$	313
10^{-5}	$[1,4,4,1]$	460	$[1,4,2,1]$	331
10^{-6}	$[1,5,5,1]$	431	$[1,5,2,1]$	317
10^{-7}	$[1,6,6,1]$	465	$[1,6,2,1]$	302

For the wrong ordering $(1,2,3)$ the ranks keep increasing as ε is decreased. This indicates that the high numerical rank is being propagated through dimension 2. On the contrary, with the ordering $(1,3,2)$, only the first of the TT-ranks increases with the decrease of ε, whereas the TT-rank between dimensions 3 and 2 is constant, having (5.54) an analytical rank-two representation with respect to these two variables. The effect of selecting the wrong ordering is also evident from the count of function evaluations.

Here we propose a novel greedy algorithm for the approximation of the best ordering, leading to the minimum TT-ranks. It builds up on the ideas used in the cut-HDMR decomposition presented in section 5.3, and it thus suffer of the same problems regarding the choice of an anchor point. The strategy will be presented through a simple example.

Example 5.9 (Second-order ordering strategy) The function

$$f(\mathbf{x}) = \frac{1}{x_1 + x_2 + 1} \qquad \mathbf{x} \in S \equiv [0,1]^2 \tag{5.55}$$

has no analytical low-rank representation of the form (5.51). This is a particular case of the "corner peak" Genz function in two dimensions. We will use this function as a building block for a function which DTT/FTT-decomposition will be sensitive to the ordering.

Let $\{M_i\}_{i=1}^k$ such that $\sum_{i=1}^k M_i = d_\mathrm{s}$ and let $\mathcal{M} = \left\{ \left\{ i_j^{(1)} \right\}_{j=1}^{M_1}, \ldots, \left\{ i_j^{(k)} \right\}_{j=1}^{M_k} \right\}$ be an arbitrary partition of $[1,2,\ldots,d_\mathrm{s}]$. In the same fashion of the Genz functions (5.46), let $\{c_i \sim \mathcal{U}([0,1])\}_{i=1}^{d_\mathrm{s}}$. Then we define the QoI function to be

$$f(\mathbf{x}) = \overbrace{\prod_{l=1}^{k} \underbrace{\left(1 + \sum_{j=1}^{M_l} c_{i_j^{(l)}} x_{i_j^{(l)}}\right)^{-1}}_{\substack{\text{High-rank intra the} \\ \text{partition sets } \left\{i_j^{(l)}\right\}_{j=1}^{M_l}}}}^{\text{Low-rank in between partitions}} . \tag{5.56}$$

For this example we select $d_\mathrm{s} = 20$ and we aim at the construction of the STT-decomposition of polynomial order 7. This means that the candidate points of the Gauss cubature rule are 8^{20}. We generate the partition[a]

$$\begin{aligned}\mathcal{M} = \{&\{3, 16, 6\}, \{10, 2, 14, 4, 17\}, \{7, 1, 13\}, \\ &\{0, 19, 18, 9, 15\}, \{8, 12, 11, 5\}\} \ ,\end{aligned} \tag{5.57}$$

where we also denote by $\mathcal{T} = \{t_j\}_{j=1}^{d_\mathrm{s}}$ the concatenated entries of $\mathcal{M}$:

$$\begin{aligned}\mathcal{T} = \{&3, 16, 6, 10, 2, 14, 4, 17, 7, 1, 13, \\ &0, 19, 18, 9, 15, 8, 12, 11, 5\} \ .\end{aligned} \tag{5.58}$$

Of course f assumes the natural order of the dimensions $(0, \ldots, 19)$ and the partition $\mathcal{M}$ is only used internally. A direct approximation of f without reordering the dimensions turns out to be computationally very expensive and we are not going to attempt its construction. Instead we will approximate $\mathcal{T}$ with $\tilde{\mathcal{T}}$ and construct the approximation of the function $\tilde{f}$ defined by:

$$\tilde{f}(x_{\tilde{t}_1}, \ldots, x_{\tilde{t}_{d_\mathrm{s}}}) = f(x_1, \ldots, x_{d_\mathrm{s}}) \tag{5.59}$$

We will proceed in several steps.

Construction of the vicinity matrix.
Let $(\hat{x}_1, \ldots, \hat{x}_{d_\mathrm{s}}) = \hat{\mathbf{x}} \in \boldsymbol{\mathcal{X}}$ be an anchor point and define

$$f_{i,j}(x_i, x_j) = f(\hat{x}_1, \ldots, \hat{x}_{i-1}, x_i, \hat{x}_{i-1}, \ldots, \hat{x}_{j-1}, x_j, \hat{x}_{j+1}, \ldots, \hat{x}_{d_\mathrm{s}}) \ . \tag{5.60}$$

In this example we chose $\hat{\mathbf{x}}$ to be one of the points in $\boldsymbol{\mathcal{X}}$ close to the center of the domain S. Then we estimate the numerical rank of all the matrices defined by the evaluation of $f_{i,j}$ on the subset of the candidate Gauss points $\boldsymbol{\mathcal{X}}$ along the planes passing through $\hat{\mathbf{x}}$. These ranks can be estimated by the matrix-SVD, which requires the evaluation of all the entries along the planes, or they can be approximated using the `TT-dmrg-cross` algorithm on the *quantic* folding of these planes. We will call these ranks *second order ranks* and collect them in the $d_\mathrm{s} \times d_\mathrm{s}$ matrix $\mathbf{R}$. Figure 5.17a shows the second order ranks of the function

f. We can define the *vicinity matrix* $\mathbf{V}$ by

$$\mathbf{V}_{i,j} = \begin{cases} \frac{1}{\mathbf{R}_{i,j}} & i \neq j \\ 0. & i = j \end{cases} \tag{5.61}$$

Traveling Salesman Problem.
The following conjecture is used in the following in order to state the ordering problem in terms of the vicinity matrix.

Conjecture 5.1 *If the approximation of the second order ranks is constant with respect to the anchor point, then the shortest path connecting all the nodes of the graph defined by the vicinity matrix defines the ordering with minimum TT-ranks.*

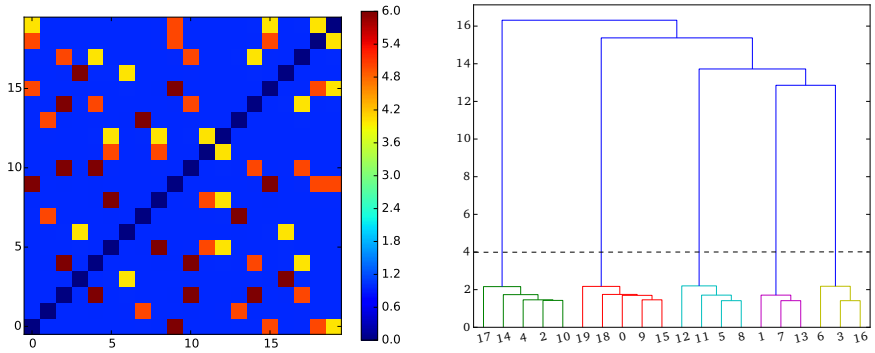
The problem of finding the shortest path connecting all the nodes of the graph defined by the vicinity matrix is known as the *Traveling Salesman Problem* (TSP) [141] and is NP-hard. Several algorithms have appeared in the last 50 years to solve this problem using different heuristics. The study of these algorithms along with their application to the tensor-train ordering problem is still under active research.

In this example however we want to show that an approximate ordering can also be useful for the problem at hand. Instead of solving the TSP, we solve the relaxed problem of clustering the nodes of the graph defined by the vicinity matrix. We use a hierarchical clustering [32] on the vicinity matrix and we obtain the dendrogram shown in figure 5.17b. We manually select a truncation on the dendrogram at level 4, obtaining the partition

$$\begin{aligned} \tilde{\mathcal{M}} = \{&\{2, 4, 10, 14, 17\}, \{0, 9, 15, 18, 19\}, \\ &\{5, 8, 11, 12\}, \{1, 7, 13\}, \{3, 6, 16\}\} \ . \end{aligned} \tag{5.62}$$

Despite not having the same order of $\mathcal{M}$ in (5.57), this partition identifies the dimensions which must be kept contiguous.

Approximation of the re-ordered function.
Using the partition $\tilde{\mathcal{M}}$, the function $\tilde{f}$ can now be approximated by the STT-decomposition. Since $\tilde{f}$ and f are related by (5.59), the approximation of $\tilde{f}$ leads to the approximation of f. The L^2 error of this approximation is then checked against the L^2 error of the approximation obtained using the exact ordering $\mathcal{M}$ and they are found both to be of the order 10^{-5}. The benefit of



(a) Second order ranks (b) Clustering

Figure 5.17: Second-order ordering strategy. Left: Matrix of second-order ranks from which the vicinity matrix will be constructed. Right: clustering of the nodes of the graph defined by the vicinity matrix.

the usage of a correct ordering is shown by the TT-ranks of the approximation:

$$\mathbf{r} = [\mathbf{1}, 8, 10, 9, 8, \mathbf{2}, 8, 8, 7, 6, \mathbf{1}, 7, 7, 5, \mathbf{1}, 7, 7, \mathbf{1}, 6, 6, \mathbf{1}] \,,$$

where we have highlighted the ranks between each cluster. These ranks are small as expected from the definition (5.52) of f.

[a]Here and in the following the counting of the dimensions is started from 0, in accordance with the implementation.

Re-weighted DTT-decomposition. Given the tensor $\boldsymbol{\mathcal{A}} = f(\boldsymbol{\mathcal{X}})$ the DTT-decomposition is based on the optimization problems stated in definition 5.2:

"find $\boldsymbol{\mathcal{A}}_{TT}$ in the format (5.32) such that $\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_{TT}\|_F \leq \varepsilon \|\boldsymbol{\mathcal{A}}\|_F$". (5.63)

The same optimality condition is seeked whether one uses the `TT-SVD` algorithm or the `TT-dmrg-cross` algorithm.

The optimization (5.63) with respect to the Frobenious norm is not satisfactory from the UQ perspective, where often there are parts of the domain which are more relevant than other. In general one seek the best approximation with respect to the $L^2_\pi(\mathbb{R}^{d_s})$ norm – see (B.10). To this end, let $\boldsymbol{\mathcal{W}} = \mathbf{w}_1 \circ \ldots \circ \mathbf{w}_{d_s}$ be the tensor containing the Gauss-type weights of the tensorized grid $\boldsymbol{\mathcal{X}}$, associated to the product measure π. Now, let $h(\boldsymbol{\mathcal{X}}_\mathbf{i}) = f(\boldsymbol{\mathcal{X}}_\mathbf{i})\sqrt{\boldsymbol{\mathcal{W}}_\mathbf{i}}$. For $\boldsymbol{\mathcal{B}} = h(\boldsymbol{\mathcal{X}})$, one can seek the DTT-decomposition $\boldsymbol{\mathcal{B}}_{TT}$ satisfying $\|\boldsymbol{\mathcal{B}} - \boldsymbol{\mathcal{B}}_{TT}\|_F \leq$

$\varepsilon\|\boldsymbol{\mathcal{B}}\|_F$. Without loss of generality, let all the tensors involved be in $\mathbb{R}^{n\times\cdots\times n}$, and note that

$$\begin{aligned}\|\boldsymbol{\mathcal{B}}-\boldsymbol{\mathcal{B}}_{TT}\|_F^2 &= \sum_{|\mathbf{i}|_0=1}^{n}\left(h(\boldsymbol{\mathcal{X}}_{\mathbf{i}})-h_{TT}(\boldsymbol{\mathcal{X}}_{\mathbf{i}})\right)^2\\ &= \sum_{|\mathbf{i}|_0=1}^{n}\left(f(\boldsymbol{\mathcal{X}}_{\mathbf{i}})-f_{TT}(\boldsymbol{\mathcal{X}}_{\mathbf{i}})\right)^2\boldsymbol{\mathcal{W}}_{\mathbf{i}} = \|f-f_{TT}\|_{L^2_\pi(\mathbb{R}^{d_\mathrm{s}})}\,, \qquad (5.64)\\ \|\boldsymbol{\mathcal{B}}\|_F^2 &= \sum_{|\mathbf{i}|_0=1}^{n} h^2(\boldsymbol{\mathcal{X}}_{\mathbf{i}}) = \sum_{|\mathbf{i}|_0=1}^{n} f^2(\boldsymbol{\mathcal{X}}_{\mathbf{i}})\boldsymbol{\mathcal{W}}_{\mathbf{i}} = \|f\|_{L^2_\pi(\mathbb{R}^{d_\mathrm{s}})}\,.\end{aligned}$$

Thus, the application of `TT-SVD` or `TT-dmrg-cross` to $\boldsymbol{\mathcal{B}}$ will result in the construction of the approximate tensor $\boldsymbol{\mathcal{B}}_{TT}$ such that $\|f-f_{TT}\|_{L^2_\pi(\mathbb{R}^{d_\mathrm{s}})} \leq \varepsilon\|f\|_{L^2_\pi(\mathbb{R}^{d_\mathrm{s}})}$, which is the right norm to be used in the UQ context. The tensor $\boldsymbol{\mathcal{A}}_{TT}$ can then be recovered from $\boldsymbol{\mathcal{A}}_{TT}=\boldsymbol{\mathcal{B}}_{TT}/\sqrt{\boldsymbol{\mathcal{W}}}$.

Anisotropic adaptivity. The algorithm `TT-dmrg-cross` provides already an adaptive mechanism for the selection of the important evaluation points for the construction of the DTT-decomposition $\boldsymbol{\mathcal{A}}_{TT}\simeq f(\boldsymbol{\mathcal{X}})$. The construction of the STT-decomposition is then obtained by projection and the core element of this projection is the tensor of generalized Fourier coefficients $\boldsymbol{\mathcal{C}}_{TT}$, which is already in the DTT-decomposition format (5.32).

The QoI function f fulfills (PU-0)-(PU-3) by assumption and we additionally assume $f\in\mathcal{H}^k_\pi(\mathbb{R}^{d_\mathrm{s}})$ for $k>d_\mathrm{s}-1$, which implies the convergence of the FTT-decomposition by theorem 5.1. These assumptions, however, do not provide any information relative to the required polynomial order of the STT-decomposition needed to fulfill a target tolerance $\varepsilon>0$ in the approximation. The choice of this polynomial order must be done adaptively and anisotropically, because the QoI function can exhibit anisotropic complexity – i.e. different complexity along different dimensions.

To this end, let $\mathbf{N}=(n_1,\ldots,n_{d_\mathrm{s}})$ and $\mathbf{M}=(m_1,\ldots,m_{d_\mathrm{s}})$ be two multi-indices such that $\mathbf{N}<\mathbf{M}$, with the meaning $n_i<m_i$ for all $i\in[1,\ldots,d_\mathrm{s}]$. Let us define the two projection operators $\mathcal{P}_{\mathbf{N}}: L^2_\pi\to\operatorname{span}\left(\{\Phi_{\mathbf{j}}\}_{\mathbf{j}<\mathbf{N}}\right)$ and $\mathcal{P}_{\mathbf{M}}: L^2_\pi\to\operatorname{span}\left(\{\Phi_{\mathbf{j}}\}_{\mathbf{j}<\mathbf{M}}\right)$. Assuming exact inner products, using the orthonormality of $\{\Phi_{\mathbf{j}}\}_{\mathbf{j}<\mathbf{M}}$ and by the definition (5.9) of the projection operators, we have

$$\|\mathcal{P}_{\mathbf{N}}f_{TT}-\mathcal{P}_{\mathbf{M}}f_{TT}\|^2_{L^2_\pi} = \sum_{\mathbf{i}=\mathbf{N}}^{\mathbf{M}} c_{\mathbf{i}}^2\|\Phi_{\mathbf{i}}\|^2_{L^2_\pi} = \sum_{\mathbf{i}=\mathbf{N}}^{\mathbf{M}} c_{\mathbf{i}}^2\,, \qquad (5.65)$$

where $c_{\mathbf{i}}$ are the generalized Fourier coefficients forming the tensors $\boldsymbol{\mathcal{C}}^{\mathbf{N}}$ and $\boldsymbol{\mathcal{C}}^{\mathbf{M}}$.

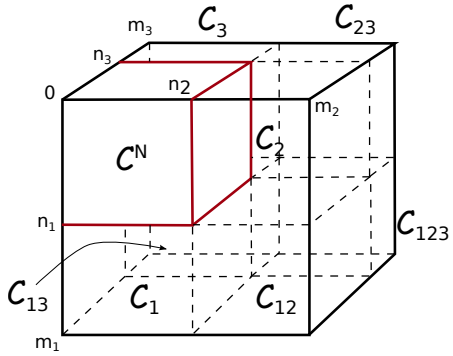


Figure 5.18: Splitting of the tensor $\mathcal{C}^{\mathbf{M}}$ of generalized Fourier coefficients.

Now, let us define

$$\mathcal{C}_{\mathbf{j}} = \mathcal{C}^{\mathbf{M}}|_{\mathbf{I}}\,, \qquad \mathbf{I} = \left\{ \mathbf{i} : \begin{cases} n_k + 1 \le i_k \le m_k & \text{if } k \in \mathbf{j} \\ 0 \le i_k \le n_k & \text{if } k \notin \mathbf{j} \end{cases} \right\} \tag{5.66}$$

where $\mathcal{C}^{\mathbf{M}}|_{\mathbf{I}}$ denotes the restriction of tensor $\mathcal{C}^{\mathbf{M}}$ to the indices in $\mathbf{I}$ and $\mathbf{j} = (j_1, \dots, j_k)$ is a multi-index with $k \le d_{\mathrm{s}}$. See figure 5.18 for a graphical definition of $\mathcal{C}_{\mathbf{j}}$ for $d_{\mathrm{s}} = 3$. From (5.65) and assuming exact inner products, we have

$$\begin{aligned} \|\mathcal{P}_{\mathbf{N}} f_{TT} - \mathcal{P}_{\mathbf{M}} f_{TT}\|_{L^2_\pi} &= \left\| \bigoplus_{\#\mathbf{i}=1} \mathcal{C}_{\mathbf{i}} \oplus \bigoplus_{\#\mathbf{i}=2} \mathcal{C}_{\mathbf{i}} \oplus \cdots \oplus \mathcal{C}_{(1,\dots,d_{\mathrm{s}})} \right\|_F \\ &= \sqrt{\sum_{\#\mathbf{i}=1} \|\mathcal{C}_{\mathbf{i}}\|_F^2 + \sum_{\#\mathbf{i}=2} \|\mathcal{C}_{\mathbf{i}}\|_F^2 + \cdots + \|\mathcal{C}_{(1,\dots,d_{\mathrm{s}})}\|_F^2}\,, \end{aligned} \tag{5.67}$$

where $\#\mathbf{i} = k$ indicates all the multi indices of size $k \le d_{\mathrm{s}}$ belonging to $\{(i_1, \dots, i_k) : i_j \in [1, \dots, d_{\mathrm{s}}] \text{ and } i_1 < \dots < i_k\}$. Since the inner products are approximated by discrete quadratures, also the core tensor $\mathcal{C}^{\mathbf{N}}$ will be different from $\mathcal{C}_0^{\mathbf{M}} := \mathcal{C}^{\mathbf{M}}|_{\mathbf{I}}$, where $\mathbf{I} = \{\mathbf{i} : i_k \le n_k,\ \forall k \in [1, \dots, d_{\mathrm{s}}]\}$. Then the total error needs to be adjusted with the error introduced by the quadratures:

$$\left\|\tilde{\mathcal{P}}_{\mathbf{N}} f_{TT} - \tilde{\mathcal{P}}_{\mathbf{M}} f_{TT}\right\|_{L^2_\pi} = \sqrt{\|\mathcal{C}^{\mathbf{N}} - \mathcal{C}_0^{\mathbf{M}}\|_F^2 + \sum_{\#\mathbf{i}=1} \|\mathcal{C}_{\mathbf{i}}\|_F^2 + \cdots + \|\mathcal{C}_{(1,\dots,d_{\mathrm{s}})}\|_F^2}\,. \tag{5.68}$$

Using these observations, we define the error contribution in direction $j \in$

$[1, \ldots, d_{\mathrm{s}}]$ to be

$$\mathscr{E}_j = \left(\|\boldsymbol{\mathcal{C}}_j\|_F^2 + \sum_{\substack{\#\mathbf{i}=2 \\ j\in\mathbf{i}}} \|\boldsymbol{\mathcal{C}}_{\mathbf{i}}\|_F^2 + \sum_{\substack{\#\mathbf{i}=3 \\ j\in\mathbf{i}}} \|\boldsymbol{\mathcal{C}}_{\mathbf{i}}\|_F^2 + \cdots + \|\boldsymbol{\mathcal{C}}_{(1,\ldots,d_{\mathrm{s}})}\|_F^2 \right)^{\frac{1}{2}} . \tag{5.69}$$

Thus the error in one direction is estimated considering all the errors which involve the refinement – the increase of polynomial order from $\mathbf{N}$ to $\mathbf{M}$ – in this direction. As we will see in section 5.3 and in chapter 6, this definition is similar to the ANOVA splitting of the variance. In practice both $\boldsymbol{\mathcal{C}}^{\mathbf{N}}$ and $\boldsymbol{\mathcal{C}}^{\mathbf{M}}$ are approximated by $\boldsymbol{\mathcal{C}}^{\mathbf{N}}_{TT}$ and $\boldsymbol{\mathcal{C}}^{\mathbf{M}}_{TT}$ in the DTT-decomposition format. The estimation of the $\boldsymbol{\mathcal{C}}_{\mathbf{j}}$'s in (5.66) can be rapidly obtained in the DTT-decomposition format from the DTT-decompositions $\boldsymbol{\mathcal{C}}^{\mathbf{N}}_{TT}$ and $\boldsymbol{\mathcal{C}}^{\mathbf{M}}_{TT}$, by a truncation of the corresponding modes[7] – see (5.66) and figure 5.18. The Frobenious norm of tensors in the DTT-decomposition format is a computationally cheap operation – $\mathcal{O}(d_{\mathrm{s}} n r^3)$ [122]. However for $d_{\mathrm{s}} \gg 1$, the summation (5.69) includes an exponentially increasing number of summands. Thus we define the n-th order error contribution in the j-th direction to be

$$\mathscr{E}_j^{(n)} = \left(\|\boldsymbol{\mathcal{C}}_j\|_F^2 + \sum_{\substack{\#\mathbf{i}=2 \\ j\in\mathbf{i}}} \|\boldsymbol{\mathcal{C}}_{\mathbf{i}}\|_F^2 + \cdots + \sum_{\substack{\#\mathbf{i}=n \\ j\in\mathbf{i}}} \|\boldsymbol{\mathcal{C}}_{\mathbf{i}}\|_F^2 \right)^{\frac{1}{2}} . \tag{5.70}$$

On the base of (5.70), different strategies for the selection of the directions to be refined are possible. One strategy is to use a *one-at-a-time* approach, where the direction of refinement is

$$j = \arg\max_j \mathscr{E}_j^{(n)} . \tag{5.71}$$

An alternative is to allow the refinement of multiple dimensions at a time. For example a *cut-off* approach is to define $\mathscr{E}_{\max}^{(n)} = \max_j \mathscr{E}_j^{(n)}$ and $\mathscr{E}_{\min}^{(n)} = \min_j \mathscr{E}_j^{(n)}$ and refine the directions $\mathbf{j}$ such that for all $j \in \mathbf{j}$,

$$\log_{10} \mathscr{E}_j^{(n)} \geq \log_{10} \mathscr{E}_{\max}^{(n)} - \alpha \left| \log_{10} \mathscr{E}_{\max}^{(n)} - \log_{10} \mathscr{E}_{\min}^{(n)} \right| , \tag{5.72}$$

where $0 \leq \alpha \leq 1$. For $\alpha = 0$ this strategy corresponds to the *one-at-a-time* strategy – maximum anisotropy –, while for $\alpha = 1$ this strategy refines all the directions at once – minimum anisotropy.

[7] In the multi-linear algebra terminology, the "mode" or "way" i is what we have been calling the dimension/direction i.

The stopping criterion can be based on the error in the estimation of the core $\left\|\boldsymbol{\mathcal{C}}^{\mathbf{N}}_{TT} - \boldsymbol{\mathcal{C}}^{\mathbf{M}}_{0,TT}\right\|_F$ with respect to a threshold $\varepsilon_{\text{ad}} > 0$. A more accurate estimate can be based on the approximation of the total error (5.68), reusing the partial contributions $\boldsymbol{\mathcal{C}}_{\mathbf{j}}$ – c.f. (5.66) – already computed for (5.70). The n-th order estimate of the total error is then

$$\boldsymbol{\mathcal{E}} = \left(\|\boldsymbol{\mathcal{C}}^{\mathbf{N}} - \boldsymbol{\mathcal{C}}^{\mathbf{M}}_0\|^2_F + \sum_i \|\boldsymbol{\mathcal{C}}_i\|^2_F + \cdots + \sum_{\#\mathbf{j}=n} \|\boldsymbol{\mathcal{C}}_{\mathbf{j}}\|^2_F \right)^{\frac{1}{2}} . \tag{5.73}$$

The anisotropic adaptivity needs to be carefully implemented in order not to waste already computed information. One needs first to chose appropriate nested quadrature rules [99–103], in order to have any hope of reusing previous computations. Unlike other approximation techniques, where a refinement determines the evaluation of the QoI on some *a priori* known points, the STT-decomposition based on the `TT-dmrg-cross` algorithm looks for the optimal points out of a set of candidate points. This means that there is no guarantee of reusing already computed function evaluations, unless the refinement is implemented carefully. Furthermore, a refinement can also cause the increase of the TT-ranks, and this is not know *a priori*. Different techniques for the initialization of a refined approximation are the topic of ongoing research.

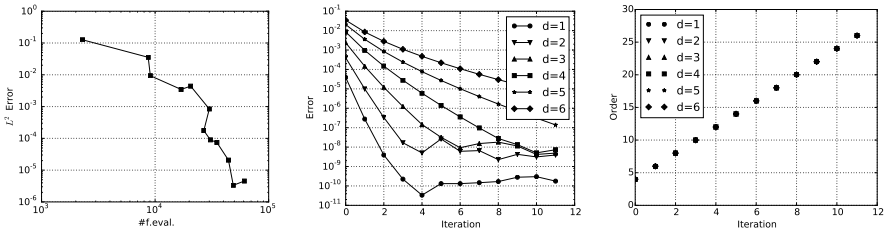
In the following example we present the anisotropic adaptivity strategy, where we use second order error estimators to determine the refinement directions and the stopping criteria. The correct implementation of the strategy is still under investigation, thus we will not use the nested rules and restarting strategies discussed before. However, the example helps showing the benefits obtained by the adoption of anisotropic adaptivity.

Example 5.10 (Anisotropic adaptivity) Let $d_{\text{s}} = 6$ and the QoI function be defined by

$$f(\mathbf{x}) = \left(1 + \sum_{i=1}^{d_{\text{s}}} x_i e^{i-1} \right)^{-1} , \tag{5.74}$$

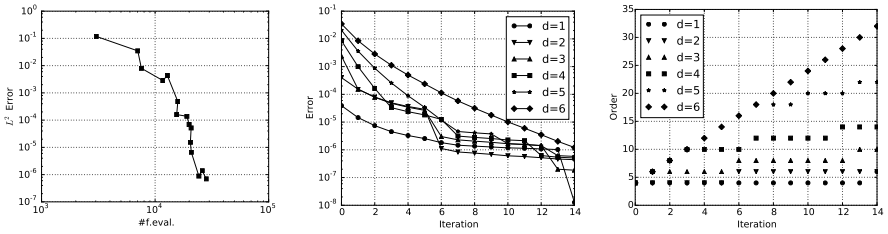
where $(x_1, \ldots, x_{d_{\text{s}}}) = \mathbf{x} \in S \equiv [0,1]^{d_{\text{s}}}$. The function f decays slowly with respect to the first dimensions, and very quickly with respect to the higher dimensions.

This example does not involve any re-usage strategy of the already computed estimates during the refinement steps[a]. Thus, we construct the STT-decomposition using Gauss-type quadrature rules (not nested). The aim of this example is the sole investigation of different adaptivity patterns. We use second order error estimators for both the selection of the anisotropic refinement directions and



(a) Approximation errors (b) 2-nd order errors (c) Polynomial orders

Figure 5.19: Example 5.10: Isotropic adaptivity for the construction of the STT-decomposition of (5.74). Left: evolution of the L^2 error in the approximation with respect to the number of samples needed for increasing polynomial orders. Center: evolution of the directional errors (5.70) with respect to the refinement iterations. Right: polynomial order in the different directions with respect to the refinement iterations.



(a) Approximation errors (b) 2-nd order errors (c) Polynomial orders

Figure 5.20: Example 5.10: Anisotropic adaptivity with linear increase of the polynomial order for the construction of the STT-decomposition of (5.74). Left: evolution of the L^2 error in the approximation with respect to the number of samples needed for increasing polynomial orders. Center: evolution of the directional errors (5.70) with respect to the refinement iterations. Right: polynomial order in the different directions with respect to the refinement iterations.

the error approximation – $n = 2$ in (5.70) and (5.73). We consider three kinds of refinements:

- *Linear isotropic*: the polynomial order is increased linearly – by steps of 2 – along all the directions at each refinement step. This strategy corresponds to the `cut-off` strategy (5.72) with $\alpha = 1$.
- *Linear anisotropic*: we use the *cut-off* strategy (5.72) with $\alpha = 0.5$. The polynomial order is increased linearly – by steps of 2 – in the directions which need refinement.
- *Exponential anisotropic*: we use the *cut-off* strategy (5.72) with $\alpha = 0.1$. The polynomial order is increased exponentially – it is doubled each time – in the directions which need refinement.

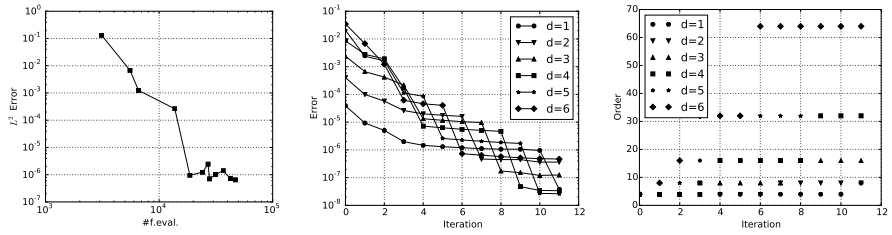
Figures 5.19, 5.20 and 5.21 show the performances of these different strategies. Figures 5.19a, 5.20a and 5.21a show the error L^2 error (5.47) estimated using the LHC method. The isotropic strategy requires approximately three times the number of function evaluations required from both the linear and exponential anisotropic strategies in order to almost reach the same accuracy. The final ranks of the three estimates are approximately the same:

Linear isotropic	$[1, 5, 7, 9, 10, 10, 1]$
Linear anisotropic	$[1, 5, 7, 10, 10, 10, 1]$
Exponential anisotropic	$[1, 6, 9, 10, 10, 11, 1]$

This means that the gain due to the anisotropic adaptivity is only caused by the lower number of candidate points in the DTT-decomposition via `TT-dmrg-cross`. This also suggests that this gain would increase rapidly with d_{s} due to exponential growth of the candidate points. The flattening of the approximation around 10^{-6} in the linear and exponential cases is due to the tolerance $\varepsilon = 10^{-6}$ set in the `TT-dmrg-cross` algorithm – c.f. (5.63).

Figures 5.19b, 5.20b and 5.21b show the decay of the directional contributions (5.70) to the error. While for the isotropic adaptivity in figure 5.19b small errors decrease even more rapidly than big errors, for the anisotropic adaptivity in figures 5.20b and 5.21b the big errors are tackled first. In fact, the anisotropic adaptivity aims at making the directional contributions to the error converge.

Finally, figures 5.19c,5.20c and 5.21c show the polynomial orders used for each direction. While in the isotropic adaptivity case all the orders are increased at each iteration, figures 5.20c and 5.21c show that in the anisotropic case the last direction is the one on which most of the refinement is applied, accordingly



(a) Approximation errors (b) 2-nd order errors (c) Polynomial orders

Figure 5.21: Example 5.10: Anisotropic adaptivity with exponential increase of the polynomial order for the construction of the STT-decomposition of (5.74). Left: evolution of the L^2 error in the approximation with respect to the number of samples needed for increasing polynomial orders. Center: evolution of the directional errors (5.70) with respect to the refinement iterations. Right: polynomial order in the different directions with respect to the refinement iterations.

with the definition (5.74) of the QoI function.

[a] This is a functionality under active development in [Bigoni, 11].

5.3 High dimensional model representation

Any vector valued QoI function $\mathbf{f} : S \to \mathbb{R}^n$ can be written as

$$\mathbf{f}(\mathbf{x}) \equiv \mathbf{f}_0 + \sum_i \mathbf{f}_i(\mathbf{x}_i) + \sum_{i<j} \mathbf{f}_{ij}(\mathbf{x}_i, \mathbf{x}_j) + \cdots + \mathbf{f}_{12\cdots d_\mathrm{s}}(\mathbf{x}_1, \ldots, \mathbf{x}_{d_\mathrm{s}}) \,, \qquad (5.75)$$

for $\mathbf{x} \in S$. This decomposition is called a *High Dimensional Model Representation* (HDMR) and it turns the high-dimensional function $\mathbf{f}$ into a sum of functions representing increasing parameter interactions. Decomposition (5.75) if far from being unique. The most trivial of these decomposition is obtained by taking $\mathbf{f}_0 = \mathbf{f}_i = \mathbf{f}_{ij} = \ldots = \mathbf{f}_\mathbf{i} = 0$, where $\mathbf{i}$ are multi-indices of size $d_\mathrm{s} - 1$, and letting $\mathbf{f}_{12\cdots d_\mathrm{s}} := \mathbf{f}$. The HDMR method [140, 142–146]-[Bigoni et al., 4–6] aims at the construction of an approximation (5.75) where the low-dimensional functions $\mathbf{f}_0, \{\mathbf{f}_i\}_i, \{\mathbf{f}_{ij}\}_{i<j}, \ldots$ carry most of the information[8] regarding $\mathbf{f}$ and the high-dimensional functions can be assumed to be zero, leading to a truncation of (5.75). For example a truncation including up to second order interactions is

[8] In this case the word "information" is used in an informal way and not in the context of information theory.

given by

$$\mathbf{f}(\mathbf{x}) \simeq \mathbf{f}_0 + \sum_i \mathbf{f}_i(\mathbf{x}_i) + \sum_{i<j} \mathbf{f}_{ij}(\mathbf{x}_i, \mathbf{x}_j) \ . \tag{5.76}$$

This relies on the assumption that $\mathbf{f}$ has such a form and its behavior is strongly influenced by single/couples/triples of variables. This is a pretty commonly fulfilled assumption in dynamical systems.

Without loss of generality the following theory will be presented considering the scalar function f in place of $\mathbf{f}$ and we will extend S to $\mathbb{R}^{d_s}$ by extension of the measure $\pi_{\mathbf{x}} : \mathscr{B}(S) \to \mathbb{R}$ to $\pi_{\mathbf{x}} : \mathscr{B}(\mathbb{R}^{d_s}) \to \mathbb{R}$ as done in (5.4).

In the following we will make use of assumptions (PU-1),(PU-2) and (PU-0). We will also require assumption (PU-3) for a mixed PC-HDMR approach.

By assumptions (PU-1) $\pi_{\mathbf{x}} = \prod \pi_{\mathrm{x}_i}$. We will let $f \in \mathcal{X}$, where $\mathcal{X}$ is a linear vector space of $\pi_{\mathbf{x}}$ integrable and measurable functions – c.f. appendix B – equipped with the $\pi_{\mathbf{x}}$-weighted inner product

$$(f, g)_{\pi_{\mathbf{x}}} = \int_{\mathbb{R}^{d_s}} f g \pi_{\mathbf{x}}(\,\mathrm{d}\mathbf{x}) \qquad f, g \in \mathcal{X} \ . \tag{5.77}$$

Now let the spaces $V_0, \{V_i\}, \{V_{ij}\}_{i<j}, \ldots, V_{12\ldots n} \subset \mathcal{X}$ be defined as:

$$\begin{aligned} V_0 &\equiv \{f \in \mathcal{X} : f = c, c \in \mathbb{R}\} \ , \\ V_{i_1,\ldots,i_l} &\equiv \Big\{ f \in \mathcal{X} : f(\mathbf{x}) = f_{i_1,\ldots,i_l}(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l}) \text{ and} \\ &\qquad \int_{\mathbb{R}} f(\mathbf{x}) \pi_{\mathrm{x}_k}(\,\mathrm{d}\mathbf{x}_k) = 0 \ , \ \forall k \in \{i_1, \ldots, i_l\} \Big\} \ . \end{aligned} \tag{5.78}$$

Then we have that

$$\mathcal{X} = V_0 \oplus \sum_i V_i \oplus \sum_{i<j} V_{ij} \oplus \cdots \oplus V_{12\ldots n} \tag{5.79}$$

Due to the orthogonality of the spaces $V_0, \{V_i\}, \{V_{ij}\}_{i<j}, \ldots, V_{12\ldots n}$ and (5.79), it's possible to define projection operators from the space $\mathcal{X}$ to the subspaces $V_0, \{V_i\}, \{V_{ij}\}_{i<j}, \ldots, V_{12\ldots n}$ such that the following properties are fulfilled:

1. $\mathcal{P}_{i_1,\ldots,i_l} : \mathcal{X} \to V_{i_1,\ldots,i_l}$ defines $f_{i_1,\ldots,i_l} := \mathcal{P}_{i_1,\ldots,i_l} f$ uniquely,
2. $(f_{\mathbf{i}}, f_{\mathbf{j}})_{\pi_{\mathbf{x}}} = \|f_{\mathbf{i}}\|_{\pi_{\mathbf{x}}} \delta_{\mathbf{ij}}$,
3. (5.75) contains $(2^n - 1)$ summands.

ANOVA-HDMR Now, using assumption (PU-2), let $\mathcal{X} = L^2_{\pi_{\mathbf{x}}}(\mathbb{R}^{d_s})$. With this choice, the projection operators are given by

$$
\begin{aligned}
f_0^A \equiv \mathcal{P}_0^A f(\mathbf{x}) &= \int_{\mathbb{R}^{d_s}} f(\mathbf{x}) \pi_{\mathbf{x}}(\,\mathrm{d}\mathbf{x})\,, \\
f_i^A(\mathbf{x}_i) \equiv \mathcal{P}_i^A f(\mathbf{x}) &= \int_{\mathbb{R}^{d_s-1}} f(\mathbf{x}) \prod_{i\neq j} \pi_{\mathbf{x}_j}(\,\mathrm{d}\mathbf{x}_j) - \mathcal{P}_0^A f(\mathbf{x})\,, \\
f^A_{i_1,\dots,i_l}(\mathbf{x}_{i_1},\dots,\mathbf{x}_{i_l}) \equiv \mathcal{P}^A_{i_1,\dots,i_l} f(\mathbf{x}) &= \int_{\mathbb{R}^{d_s-l}} f(\mathbf{x}) \prod_{k\notin\{i_1,\dots,i_l\}} \pi_{\mathbf{x}_k}(\,\mathrm{d}\mathbf{x}_k) \\
&\quad - \sum_{j_1<\dots<j_{l-1}\subset\{i_1,\dots,\text{ß}_l\}} \mathcal{P}^A_{j_1,\dots,j_{l-1}} f(\mathbf{x}) \\
&\quad - \dots - \sum_j \mathcal{P}_j^A f(\mathbf{x}) - \mathcal{P}_0^A f(\mathbf{x})\,.
\end{aligned}
\tag{5.80}
$$

The *ANalysis Of VAriance HDMR* (ANOVA-HDMR) expansion of f is given by:

$$
\mathbf{f}(\mathbf{x}) \equiv \mathbf{f}_0^A + \sum_i \mathbf{f}_i^A(\mathbf{x}_i) + \sum_{i<j} \mathbf{f}_{ij}^A(\mathbf{x}_i,\mathbf{x}_j) + \dots + \mathbf{f}^A_{12\dots d_s}(\mathbf{x}_1,\dots,\mathbf{x}_{d_s})\,. \tag{5.81}
$$

By construction, such expansion is exact. We will see in the following that its L-th order truncation

$$
\mathbf{f}(\mathbf{x}) \simeq \mathbf{f}^A(\mathbf{x}) \equiv \underbrace{\mathbf{f}_0^A + \sum_i \mathbf{f}_i^A(\mathbf{x}_i) + \sum_{i<j} \mathbf{f}_{ij}^A(\mathbf{x}_i,\mathbf{x}_j) + \dots}_{L\text{-th order interactions}}\,, \tag{5.82}
$$

can be used as an approximation of the QoI function. Furthermore this expansion will turn useful for the computation of sensitivity indices in section 6.1.1.

The ANOVA-HDMR approximation has a significant drawback: its evaluation requires the computation of several high-dimensional integrals, in particular for low order terms. For this reason we introduce the Cut-HDMR approximation, also known as anchored ANOVA decomposition.

Cut-HDMR Instead of computing the ANOVA-HDMR by expensive cubature rules or by pseudo-random sampling method, one can rely once more on the assumption that the function f can be represented mostly by low order nonlinear interactions, whereas high-order interactions are only additive and thus a truncation in the form (5.76) is accurate. This assumption leads to the Cut-HDMR approximation. Let $\mathbf{y} \in S$ be the *anchor point* of the approximation

and let the projection operators be defined by

$$\begin{aligned} f_0^C &\equiv \mathcal{P}_0^C f(\mathbf{x}) = f(\mathbf{y}), \\ f_i^C(\mathbf{x}_i) &\equiv P_i^C f(\mathbf{x}) = f^i(\mathbf{x}_i) - \mathcal{P}_0^C f(\mathbf{x}), \\ f_{i_1,\dots,i_l}^C(\mathbf{x}_{i_1},\dots,\mathbf{x}_{i_l}) &\equiv \mathcal{P}_{i_1,\dots,i_l}^C f(\mathbf{x}) = f^{i_1,\dots,i_l}(\mathbf{x}_{i_1},\dots,\mathbf{x}_{i_l}) - \\ &\qquad \sum_{k_1<\dots<k_{l-1}\in\{i_1,\dots,i_l\}} \mathcal{P}_{k_1,\dots,k_{l-1}}^C f(\mathbf{x}) - \\ &\qquad \dots - \sum_{k\in\{i_1,\dots,i_l\}} \mathcal{P}_k^C f(\mathbf{x}) - \mathcal{P}_0^C f(\mathbf{x}), \end{aligned} \tag{5.83}$$

where $f^{i_1,\dots,i_l}(\mathbf{x}_{i_1},\dots,\mathbf{x}_{i_l})$ is the function $f(\mathbf{x})$ with all the remaining variables set to $\mathbf{y}$ and $\mathcal{P}_{\mathbf{i}}^C$ can be interpreted as the projection operator $\mathcal{P}_{\mathbf{i}}^A$ where the measure $\pi_{\mathbf{x}}$ is substituted by the Dirac measure $\pi(\,\mathrm{d}\mathbf{x}) := \delta(\mathbf{x}-\mathbf{y})\,\mathrm{d}\mathbf{x}$. This leads to the expansion

$$\mathbf{f}(\mathbf{x}) \equiv \mathbf{f}_0^C + \sum_i \mathbf{f}_i^C(\mathbf{x}_i) + \sum_{i<j} \mathbf{f}_{ij}^C(\mathbf{x}_i,\mathbf{x}_j) + \dots + \mathbf{f}_{12\cdots d_\mathrm{s}}^C(\mathbf{x}_1,\dots,\mathbf{x}_{d_\mathrm{s}})\,, \tag{5.84}$$

which requires the evaluation of f along lines, planes and hyperplanes passing through the anchor point. An L-th order truncation of this decomposition leads to the approximation:

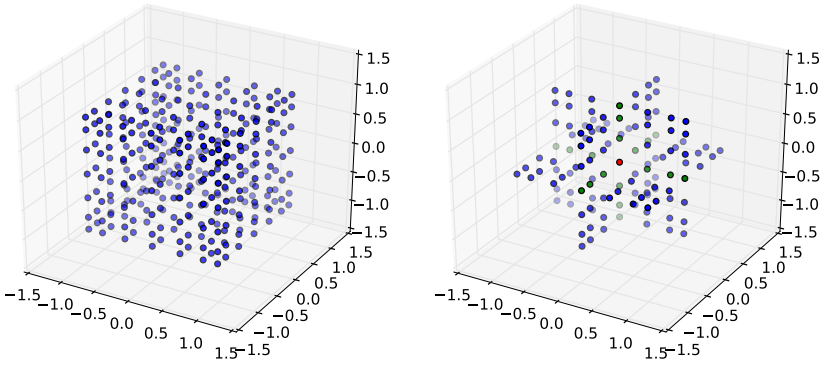
$$\mathbf{f}(\mathbf{x}) \simeq \mathbf{f}^C(\mathbf{x}) \equiv \underbrace{\mathbf{f}_0^C + \sum_i \mathbf{f}_i^C(\mathbf{x}_i) + \sum_{i<j} \mathbf{f}_{ij}^C(\mathbf{x}_i,\mathbf{x}_j) + \cdots}_{L\text{-th order interactions}}\,. \tag{5.85}$$

Now the ANOVA-HDMR decomposition (5.81) can be computed using f^C in place of f. Thanks to the form of the Cut-HDMR, the projections (5.80) involve only low-dimensional integrals. Exploiting assumption (PU-0) these can be approximated by random sampling [144] or, assuming also (PU-3), using PC based quadratures [145–147]-[Bigoni et al., 4–6]. Recent developments have also explored the usage of Tensor-Train based quadratures [140] – see section 5.2.4.2.

The construction of the ANOVA-HDMR from an M-th order PC based Cut-HDMR expansion involving interactions of order L for a function f in d_s arguments requires the evaluation of f at

$$N_\mathrm{cut} = \sum_{i=0}^{L} \binom{d_\mathrm{s}}{i} N^i \tag{5.86}$$

points, where $M = 2N+1$ is the order of the tensorized cubature rule with $N+1$ points. For $L \ll d_\mathrm{s}$, $N_\mathrm{cut} \ll (N+1)^{d_\mathrm{s}}$. Figure 5.22 shows the PC based Gauss quadrature points of order $M = 13$ for the computation of the ANOVA-HDMR from a Cut-HDMR including interactions of order $L = 2$, for a function f with $d_\mathrm{s} = 3$.



(a) Full cubature (b) Cut-HDMR

Figure 5.22: Full cubature rule and cubatures for 2-nd order interactions in Cut-HDMR, for $d_{\mathrm{s}} = 3$.

Effective Dimension It is left to determine a sufficient value of L for the approximation to be accurate. To this end we will use the concept of *effective dimension* of a function.

By assumption (PU-1), the inputs $\mathbf{x}$ consist of independently distributed random variables. Then the overall variance of f can be expressed in terms of (5.81):

$$\begin{aligned} D \equiv \mathbf{V}[f] &= \sum_i D_i + \sum_{i<j} D_{ij} + \cdots + D_{1,2,\ldots,n} \ , \\ D_{i_1,\ldots,i_l} &= \int_{\mathbb{R}^l} \left(f^A_{i_1,\ldots,i_l}\right)^2 \prod_{k\in\{i_1,\ldots,i_l\}} \pi_{\mathbf{x}_k}(\,\mathrm{d}\mathbf{x}_k) \ . \end{aligned} \tag{5.87}$$

For $0 < q \leq 1$, the effective dimension of f is $0 \leq L \leq d_{\mathrm{s}}$ such that

$$\sum_{0<|\mathbf{i}|\leq L} D_{\mathbf{i}} \geq qD \ . \tag{5.88}$$

Thus, the truncation parameter L for the ANOVA-HDMR is determined by the number of interactions which must be considered in order to express a q fraction of the total variance.

For high-dimensional problems, D can be estimated by a random sampling method or by one of the advanced method presented in section 5.2.4. Then ANOVA-HDMR approximations of increasing orders L can be constructed until the requirement (5.88) is fulfilled for a selected value of q. Note however that whether one is using a Cut-HDMR expansion based on random sampling or based on PC, both the estimate on the left hand side of (5.88) and D are approximations and their convergence needs to be checked.

Chapter 6

Sensitivity analysis

With the propagation of uncertainty we aim at the characterization of the distribution π_f. With the sensitivity analysis we investigate how the different input parameters $\mathbf{X}$ influence π_f. The goal is to identify the parameters which give the biggest contribution to the uncertainty of $f \circ \mathbf{X}$. This goal however requires a formal definition of what is uncertainty.

Traditionally, sensitivity analysis has been used in optimization problems, where the sensitivity of an objective function to its parameters is represented by its derivatives along the directions of such parameters. In this setting the uncertainty is defined in terms of the gradient: if $\nabla f(\mathbf{X}_0) = 0$ then the point $\mathbf{X}_0$ is located on a plateau of f and the uncertainty is zero, whereas if $\nabla f(\mathbf{X}_0) \gg 0$ then the point $\mathbf{X}_0$ is located on a steep plane of f and small variations of $\mathbf{X}_0$ will lead to big variations of f. In many PDE-constrained optimization problems the computation of the gradient of the objective function requires a negligible overhead when the adjoint method [148–152] is used[1]. The sensitivities used for these kind of optimization problems go under the name of *local sensitivities* because they rely on the linearization of the model around some point of interest in the space of parameters and they are only locally representative.

Local sensitivity analysis is unsatisfactory from the UQ perspective, where the probability distribution of the parameters is not necessarily narrow around their

[1]Note that the derivation of the adjoint of complex models is not always trivial, but can sometimes be achieved by automatic differentiation [153].

nominal values and thus not local. The sensitivity analysis in the UQ context must describe the sensitivity of the output distribution of the QoIs to the input distribution of the parameters. This sensitivities go under the name of *global sensitivities*. In this context the uncertainty of a model is defined by its variance, and the sensitivity to a particular input is described by the amount of total variance due to such input.

The analysis of the sensitivity of the QoI function to its inputs helps grading them according to their importance. This ranking will also allow to detect inputs which are not influential on the uncertainty of the QoI. These inputs can therefore be set to their nominal values without altering the overall uncertainty characteristics of the QoI. This technique goes under the name of *model refinement* and helps decreasing the dimensionality of the input space, leading to a model which is more manageable. Note that when sensitivity analysis is used for model refinement, its goal is primarily qualitative and secondarily quantitative. Thus, one aims to a correct ranking of the inputs rather than to the accuracy of the TSI. These accurate indices can be more easily obtained considering the refined model.

6.1 Variance-based sensitivity analysis

Variance-based sensitivity analysis [143] is the most commonly used type of global sensitivity analysis. In this context we will present the method of Sobol' [143, 144, 154] which defines indices of sensitivity as the amount of variance explained by one input and all its combinations with other inputs. This indices are said to be high-order[2] indices and we will call them *Total Sensitivity Indices* (TSIs). The first-order indices – including only variances due to single inputs – are also called *Importance measures* [143]. The TSIs are going to be obtained using the HDMR decomposition presented in section 5.3. The indices can also be found by the *Fourier amplitude sensitivity test* (FAST) method [143, 155] which is however out of the scope of this work.

6.1.1 Method of Sobol'

The method of Sobol' is very much based on the HDMR decomposition presented in section 5.3. We consider the ANOVA-HDMR decomposition of the

[2]In this context "high-order" has nothing to share with the high-order methods based on polynomial approximation presented in section 5.1.

QoI function f:

$$f(\mathbf{x}) \equiv f_0^A + \sum_i f_i^A(\mathbf{x}_i) + \sum_{i<j} f_{ij}^A(\mathbf{x}_i, \mathbf{x}_j) + \cdots + f_{12\cdots d_\mathrm{s}}^A(\mathbf{x}_1, \ldots, \mathbf{x}_{d_\mathrm{s}}) \ . \quad (5.81)$$

and we decompose the variance as

$$\begin{aligned} D \equiv \mathbf{V}[f] &= \sum_i D_i + \sum_{i<j} D_{ij} + \cdots + D_{1,2,\ldots,n} \ , \\ D_{i_1,\ldots,i_l} &= \int_{\mathbb{R}^l} \left(f_{i_1,\ldots,i_l}^A\right)^2 \prod_{k\in\{i_1,\ldots,i_l\}} \pi_{\mathrm{x}_k}(\,\mathrm{d}\mathbf{x}_k) \ . \end{aligned} \quad (5.87)$$

Then, we let

$$S_{i_1,\ldots,i_l} = \frac{D_{i_1,\ldots,i_l}}{D} \quad (6.1)$$

be the contribution of $\{X_i\}_{i\in\{i_1,\ldots,i_l\}}$ to the total variance D. The *Total Sensitivity Index* (TSI) $\mathrm{TS}(i)$ for the input X_i is defined as

$$\mathrm{TS}(i) = 1 - S_{\neg i} \ , \quad (6.2)$$

where $S_{\neg i}$ is the sum of all $S_{\mathbf{i}}$ for which $i \notin \mathbf{i}$. The TSIs do not sum to one because for $i \neq j$ there are many multi-indices $\mathbf{i}$ which contain both, but they are very useful because they allow the identification of inputs which not only affect directly the uncertainty, but which affect the uncertainty in combination with other inputs.

In order to approximate the TSI, one should first truncate the ANOVA-HDMR expansion (5.81) up to a certain level of interactions, then approximate the high-dimensional integrals in the ANOVA-HDMR expansion. To this end MC methods can be used. Alternatively, the Cut-HDMR expansion can be employed, which reduces the dimensionality of the integrals and allows the usage of PC based cubature rules.

We will use an example to show the efficacy of the TSI in highlighting the important inputs in a multivariate QoI function and the efficiency of the approach based on the Cut-HDMR expansion.

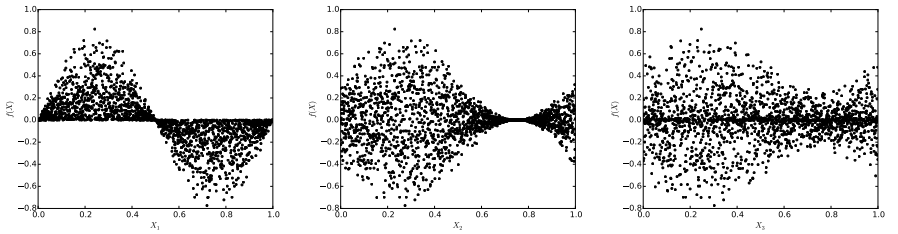
Example 6.1 Let the QoI be defined as

$$f(X) = \prod_{i=1}^{d_\mathrm{s}} \frac{\sin(2\pi X_i) + a_i}{1 + a_i} \ , \quad (6.3)$$

where $d_\mathrm{s} = 8$, $X_i \sim \mathcal{U}([0,1])$ and $a = (0, 1, 2, 4.5, 9, 20, 50, 99)$. The factor a_i determines the degree of importance of the input X_i: in fact $\frac{\sin(2\pi X_i)+a_i}{1+a_i} \to \sin(2\pi X_i)$ as $a_i \to 0$ and $\frac{\sin(2\pi X_i)+a_i}{1+a_i} \to 1$ as $a_i \to \infty$. Figure 6.1 shows the

				TSI							
L	M	N_{cut}	q	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8
1	7	17	0.26	0.275	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	11	33	0.56	0.560	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	15	49	0.56	0.557	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	7	129	0.36	0.360	0.065	0.016	0.003	0.001	0.000	0.000	0.000
	11	481	0.91	0.911	0.272	0.068	0.013	0.003	0.001	0.000	0.000
	15	1057	0.98	0.977	0.295	0.074	0.015	0.004	0.001	0.000	0.000
3	7	577	0.35	0.350	0.067	0.019	0.004	0.001	0.000	0.000	0.000
	11	4065	0.96	0.957	0.315	0.104	0.022	0.006	0.001	0.000	0.000
	15	13153	1.01	1.012	0.337	0.111	0.023	0.006	0.001	0.000	0.000
2	7	33	0.35	0.354	0.064	0.016	0.003				
	11	113	0.96	0.958	0.287	0.072	0.014				
	19	417	0.98	0.981	0.297	0.074	0.015				
3	7	65	0.35	0.352	0.067	0.019	0.004				
	11	369	1.02	1.019	0.335	0.111	0.023				
	19	2465	1.01	1.009	0.336	0.111	0.023				

Table 6.1: Example: Method of Sobol'. Results of the sensitivity analysis of (6.3), obtained using the method of Sobol' through ANOVA-HDMR and Cut-HDMR. L is the degree of interactions considered in the Cut-HDMR expansion. $M = 2N + 1$ is the polynomial degree of exactness of the Gauss quadrature rules used in the ANOVA-HDMR, where N is the number of points used for each direction. N_{cut} is the total number of function evaluations (5.86) used for the construction of the Cut-HDMR. q is the ratio between the variance expressed by the ANOVA-HDMR and the total variance $\hat{\sigma}_f$.



(a) (b) (c)

Figure 6.1: Example: Method of Sobol'. Scattering with respect to the first three inputs of the LHC points $f(\mathbf{X})$.

scattering with respect to the first three inputs of the LHC points $f(\mathbf{X})$.

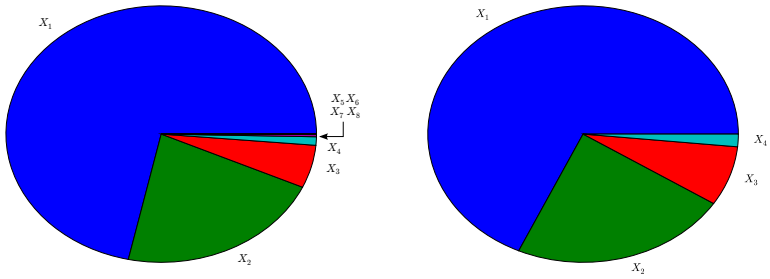
We first use the LHC method in order to estimate the total variance $\mathbf{V}[f \circ \mathbf{X}]$. We use 2000 samples, obtaining $\hat{\sigma}_f = 0.0449$, with an accuracy up to the second non-zero digit.

The ANOVA-HDMR decomposition is then constructed from several Cut-HDMR decompositions with increasing interaction orders L. PC based Gauss quadratures are employed with increasing order $M = 2N + 1$, where N is the number points used for each direction.

Table 6.1 collects the results. N_{cut} is the total number of function evaluations used for the construction of the Cut-HDMR, thus excluding the function evaluations used for the estimation of $\hat{\sigma}_f$. The ratio q between the variance expressed by the ANOVA-HDMR and the total variance $\hat{\sigma}_f$ allows the control over the effective dimensionality (5.88) of the function. We can see that $L = 1$ is not sufficient in representing the total variance, but already with $L = 2$ we achieve a satisfactory 98% of the total variance.

The TSIs show that the uncertainty in the QoI function is mostly influenced by the first input factors. Thus we refine the model by fixing the last four inputs to their nominal values and we perform a more accurate sensitivity analysis with regard to the first four inputs. The second part of table 6.1 shows such results. Due to numerical errors in both the quadratures and the LHC estimation of the total variance, the value q is slightly bigger than 1 in certain cases. However, this error is in the expected range of error given by the convergence of the LHC method for the total variance $\hat{\sigma}_f$, and moreover the qualitative results regarding the ranking of the inputs are not affected by this error.

Figure 6.2 shows a graphical interpretation of the results listed in table 6.1, where pie charts are used to represent the relative importance of the different inputs.



(a) $d_{\mathrm{s}} = 8$ (b) $d_{\mathrm{s}} = 4$

Figure 6.2: Example: Method of Sobol'. Pie plot of the TSI for (6.3).

Chapter 7

Probabilistic inverse problems

Probabilistic inverse problems are presented here for completeness of the exposition of the field of Uncertainty Quantification. A broad overview of the topic is given by setting the fundamental concepts defining the problem. Probabilistic inverse problems will not be used in the practical applications presented in part II, and thus we will not delve very deep into the topic.

In many problems in science and engineering, one is exposed to the observation of the outputs $\mathbf{Y} \in \mathbb{R}^n$ of a system without knowing its inputs $\mathbf{X} \in \mathbb{R}^{d_s}$. If the *forward* mathematical model $f : \mathbb{R}^{d_s} \to \mathbb{R}^n$ is assumed to model the physical phenomena relating $\mathbf{X}$ and $\mathbf{Y}$, then the problem

$$\begin{aligned} &\text{“Given the forward model } f \text{ and the observations } \mathbf{Y}^{\text{obs}}, \\ &\qquad \text{find the inputs } \mathbf{X}^* \text{ such that } \mathbf{Y}^{\text{obs}} = f(\mathbf{X}^*)\text{”} \end{aligned} \tag{7.1}$$

defines an *inverse problem*. Inverse problems can be over-determined if $n > d_s$ – big data – or under-determined if $n < d_s$ – small data. While over-determined problems often suffer the lack of existence of a solution satisfying all the outputs, under-determined problems often suffer the lack of uniqueness of a solution satisfying the outputs. Furthermore, the forward model can be very sensitive to its inputs, leading to difficulties in inferring them from its outputs. Inverse problems are said to be *ill-posed* if they lack existence, uniqueness or if they are highly sensitive to their inputs. Techniques used for solving ill-posed inverse problems go under the name of *regularization* techniques.

In UQ we usually work with severely under-determined problems $n \ll d_{\mathrm{s}}$, where the forward model is sometimes very sensitive to its inputs. If the forward model is exact – i.e. it represents exactly the physical model generating the observations – and the observations are noiseless, there must exist at least one solution satisfying the observations. However, the fact that the problem is under-determined and the forward model is sensitive to its inputs makes finding such a solution very difficult. Furthermore, in all practical cases the observations are not noiseless either. Then many solutions possibly satisfy the observations within the maximum accuracy allowed by the noise. For these reasons inverse problems in UQ are usually tackled using probabilistic methods.

In broad terms, the deterministic inverse problem (7.1) is rephrased into the *probabilistic inverse problem*:

$$\begin{aligned} &\text{“Given the forward model } f\text{, the noise function } g \text{ and} \\ &\qquad \text{the observations } \mathbf{Y}^{\mathrm{obs}} = g(f(\mathbf{X}))\text{, find } \mathbf{X} \text{ which} \\ &\qquad\qquad \text{is } \textit{likely} \text{ to have generated } \mathbf{Y}^{\mathrm{obs}}\text{”}. \end{aligned} \tag{7.2}$$

The term “likely” should be substituted by an optimality condition for $\mathbf{X}$ with respect to a discrepancy function between the forward model output $\mathbf{Y} = f(\mathbf{X})$ and the observations $\mathbf{Y}^{\mathrm{obs}}$.

This function is known as the *likelihood* and denoted by $L(\mathbf{X}) \equiv p_{\mathbf{y}|\mathbf{x}}(\mathbf{Y} = \mathbf{Y}^{\mathrm{obs}}|\mathbf{X}) = h(\mathbf{Y}^{\mathrm{obs}} - f(\mathbf{X}))$, for some function h which weights the misfit between $\mathbf{Y}^{\mathrm{obs}}$ and $\mathbf{Y}$. A common assumption for the noise function g is that it is additive with mutually independent and identically distributed components, i.e.:

$$\mathbf{Y} = g(f(\mathbf{X})) = f(\mathbf{X}) + \boldsymbol{\eta}\,, \tag{7.3}$$

where $\boldsymbol{\eta}$ are i.i.d. random variables with distribution π_η. If the distribution π_η has density ρ_η, then the likelihood function is defined by

$$L(\mathbf{X}^*) \equiv p_{\mathbf{y}|\mathbf{x}}(\mathbf{Y} = \mathbf{Y}^{\mathrm{obs}}|\mathbf{X} = \mathbf{X}^*) := \prod_{i=1}^{n} \rho_\eta(\mathbf{Y}_i^{\mathrm{obs}} - f_i(\mathbf{X}^*)) \tag{7.4}$$

In the following we will present the two main formulations, based on the likelihood function, for the solution of the probabilistic inverse problem (7.2). The reader is referred to the literature therein for further details.

Maximum likelihood. One approach to the solution of the probabilistic inverse problem (7.2) is to maximize the likelihood function:

$$\mathbf{X}^{\max} = \max_{\mathbf{X}} L(\mathbf{X})\,. \tag{7.5}$$

This approach goes under the name of *maximum likelihood* [32]. The maximum likelihood problem could be tackled directly with deterministic optimization techniques. However these techniques suffer the ill-posedness of the problem – the problem is under-determined and the likelihood is defined in terms of the forward model which can be nonlinear and highly sensitive with respect to the inputs – by getting stuck in local-maxima. Then probabilistic techniques [63, 65, 66] are needed to solve the problem.

Bayesian inference. One should notice that $\mathbf{Y}$ in (7.3) is a random variable with distribution $\pi_{\mathbf{y}}$ and so it is $\mathbf{X}$ if we seek it solving the problem (7.2). Thus it makes sense to seek the distribution $\pi_{\mathbf{x}}$ of $\mathbf{X}$ such that $(f(\mathbf{X}) + \boldsymbol{\eta}) \sim \pi_{\mathbf{y}}$.

This is the main task of Bayesian inference [156, 157]. It is based on the *Bayes rule*

$$\rho_{\mathbf{x}|\mathbf{y}}(\mathbf{X}|\mathbf{Y} = \mathbf{Y}^{\text{obs}}) \propto p_{\mathbf{y}|\mathbf{x}}(\mathbf{Y} = \mathbf{Y}^{\text{obs}}|\mathbf{X})\rho_{\mathbf{x}}(\mathbf{X}) \,, \tag{7.6}$$

where we have assumed that the PDFs $\rho_{\mathbf{x}|\mathbf{y}}$ and $\rho_{\mathbf{x}}$ exist for the distributions $\pi_{\mathbf{x}|\mathbf{y}}$ and $\pi_{\mathbf{x}}$ respectively – see appendix B.6 for more details on conditional probabilities. In Bayesian terminology, $\rho_{\mathbf{x}}$ represents the belief that one has regarding possible values of $\mathbf{X}$ *prior* to the observation of $\mathbf{Y}^{\text{obs}}$, while $\rho_{\mathbf{x}|\mathbf{y}}$ is the *posterior* belief on the possible values of $\mathbf{X}$ after the observation of $\mathbf{Y}^{\text{obs}}$. The prior and posterior beliefs are connected by the likelihood function $p_{\mathbf{y}|\mathbf{x}}(\mathbf{Y} = \mathbf{Y}^{\text{obs}}|\mathbf{X})$ – see (7.4) – which determines how informative $\mathbf{Y}$ is in the inference of $\mathbf{X}$.

At this point one should notice the analogy between the parametric methods for the quantification of source of uncertainties presented in section 4.3.1 and the machinery for probabilistic inverse problems presented here. The methods are practically the same, but for scope of their application. In section 4.3.1 one seeks the distribution $\pi_{\mathbf{x}}$ from observed values of $\mathbf{X}$, and the forward model is determined by the *a priori* selection of a family of probability distributions. This kinds of problems are known as *identification* problems. Here, instead, we seek $\pi_{\mathbf{x}}$ from observed values of $\mathbf{Y}$, where the forward model represents the physics governing the relation between $\mathbf{X}$ and $\mathbf{Y}$.

Markov Chain Monte Carlo. Few Bayesian inference problems (7.6) have analytic solutions. In general they need to be solved through simulations by means of the *Markov Chain Monte Carlo* (MCMC) method [156–158]. This is a pseudo-random sampling method – see section 5.1 – which allows the generation of *Markov chains*[1] with an invariant distribution. In Bayesian inference

[1] A Markov chain is a sequence of random variables $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots$ which fulfill the Markov property $P(\mathbf{X}^{(n+1)}|\mathbf{X}^{(1)} = \mathbf{x}^{(1)}, \ldots, \mathbf{X}^{(n)} = \mathbf{x}^{(n)}) = P(\mathbf{X}^{(n+1)}|\mathbf{X}^{(n)} = \mathbf{x}^{(n)})$. Note that, unlike the methods in section 5.1, the variables $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots$ of a Markov chain can be

the invariant distribution is the posterior probability $\pi_{\mathbf{x}|\mathbf{y}}$, and the transition probabilities of the chain are defined by the likelihood function L and the prior probability $\pi_{\mathbf{x}}$. Thus, MCMC applied on the Bayesian inference problem (7.6) allows one to generate samples from the posterior distribution $\pi_{\mathbf{x}|\mathbf{y}}$ without actually knowing it analytically.

We will not adventure any further on these topics and we refer the reader to more complete literature [156–158, 160]. We only mention that applications involving MCMC are some of the most computationally expensive due to the slow convergence of MC type approaches.

Recently, the PC framework presented in section 5.2 has been used for the acceleration of Bayesian inference in inverse problems [161, 162], where the computationally expensive evaluation of the forward model was substituted by the cheap evaluation of a PC approximation.

In the context of Bayesian inference in inverse problems, PC techniques have been used also for the identification problem [64, 65] presented in section 4.3.1. Instead of selecting an exponential family and inferring its parameters from the observations, in this case the observations are assumed to have a distribution described by a PC model, where the PC coefficients are then inferred from the observations.

dependent by definition. We refer the reader to [159] for details on Markov chains

Chapter 8

Conclusions and outlook

The presented framework for Uncertainty Quantification represents a valuable tool in the analysis of dynamical systems subject to uncertainties. These analysis have been traditionally performed using probabilistic methods, some of which have been presented in this part of the work. The computational bottleneck in UQ is often located in the computation of the forward model, which is required for every step of the UQ framework but the quantification of the input uncertainties – unless these are being characterized using probabilistic inversion. This forward model describes the physics of the problem, and despite big progresses in the field of high-performance computing, the solution of many of these models is computationally challenging. This fact limits the applicability of traditional probabilistic methods, which often require hundreds or thousands of evaluations of the forward model.

This justifies the appearance of a number of methods based on meta-models, which try to find a useful approximation of the forward model to the end of performing the analysis of interest. In this work we have investigated Polynomial Chaos methods which are based on the polynomial approximation of the forward model. These methods have recently gained a lot of popularity as a consequence of a big research effort into their expansion to the approximation of high-dimensional functions. The main contribution of the first part of this work falls into this path.

Approximation methods based on polynomials have been used in science since

Karl Weierstrass's proved the approximation theorem (1885) which bears his own name. Polynomials found large application in numerical methods during the last 50 years, in particular in the solution of physical problems modeled by PDEs. However these approximation methods had an hard time in entering the field of UQ, due to their poor performances – curse of dimensionality – on problems with the usual dimensionality found in UQ problems, which often exceed the three dimensions considered in PDE problems.

The novel Spectral Tensor-Train decomposition method – see section 5.2.4.2 and [Bigoni et al., 9] – aims at the polynomial approximation of high-dimensional functions while alleviating the curse of dimensionality typical of methods based on polynomials. This result is achieved at the expense of an additional assumption on the regularity of the approximated function, which is in practice required to belong to the Sobolev space of the same order as the dimensionality of its input.

The STT-decomposition exhibits performances on test functions which agree with the theory and are comparable and sometimes better than state of the art methods such as Sparse Grids [Bigoni et al., 9]. The method has been tested also on an elliptic PDE with random inputs in [Bigoni et al., 9], without however exploiting the anisotropy characteristic of the problem.

The method is still young and has room for improvement. Three research directions have been presented in this work which aim at solving the ordering problem characteristic of tensor-train decompositions, at providing better approximations in suitable norms and at exploiting the anisotropy of the approximated function in order to limit the number of function evaluations. All these enhancements are in the process of being incorporated in the `TensorToolbox` [Bigoni et al., 11] described in appendix D.3.

The application of the STT-decomposition for the propagation of uncertainty to engineering problems is under active investigation, in particular in the field of stochastic water wave simulation described in chapter 10. In this field we have identified several problems, involving optimization and probabilistic inversion, which would benefit from the usage of surrogate models based on the STT-decomposition.

If on one hand, many problems which complexity posed serious limits in the past, are now entering a domain of computational time for which the stochastic simulation and the UQ analysis becomes possible, thanks to recent developments in the field of high-performance computing. On the other hand new numerical methods for UQ are appearing which provide better analysis while requiring a lower computational burden. These two trends suggest that UQ will become increasingly important and applied in the future.

Part II

Applications of Uncertainty Quantification

Chapter 9

Railway Vehicle Dynamics

Nowadays railway transport is challenged with the demand of providing faster, cheaper and environmentally cleaner connections on medium distances, where aviation transport has been dominating during the last fifty years. These demands are pushing the development of high-speed trains and high-speed lines which are now allowing fast intra-continental transport.

New vehicles as well as new railway lines need to comply with strict regulatory safety standards and every vehicle needs to undergo a series of static and dynamic tests before being homologated by the regulatory authority. In the European Union (EU) tests on-track [163] are required for newly designed vehicles and, in many cases, also for modifications to old ones.

These tests are costly and often not fully representative of the possible running conditions. The advances achieved in computer aided simulation offer a cheaper and sometimes more representative approach, which could complement and in the end substitute some of the on-track tests [164, 165]. In order for this "virtual homologation" to be representative, UQ methods can be applied and standardized UQ procedures may enter the design phases of new vehicles.

In the last few years several works have been focusing on the characterization of the track irregularities, which are a major source of uncertainty for the dynamics of the vehicle. In [166] techniques for modeling non-stationary non-Gaussian dependent vector-valued random fields have been developed and used

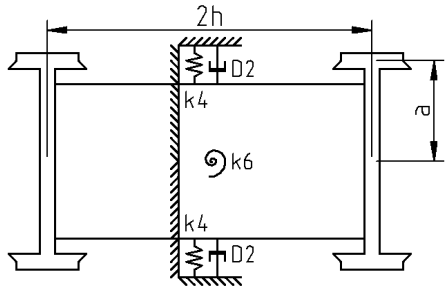


Figure 9.1: The simple Cooperrider bogie vehicle

for the characterization of track irregularities. Then, using MC simulations, the uncertainty related to these irregularities have been propagated through the dynamical systems modeling vehicles [167].

In our works [Bigoni et al., 1–6] we instead investigate the influence that the uncertainty on the characteristics of the suspension components has on the dynamic stability of vehicles. Our focus will be on the non-linear dynamics of the model under such uncertainty and we are going to neglect external excitations from track irregularities.

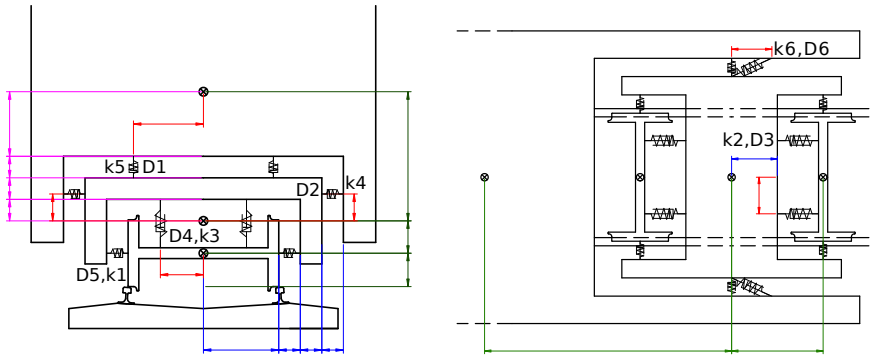
9.1 The models.

In the several works we have done on the topic, we study the dynamics of a more and more complex version of a vehicle equipped with the Cooperrider bogie [168, 169].

The simplest of such versions is shown in figure 9.1. We will call this the *simple Cooperrider bogie* vehicle. This model is composed by two conical wheel sets rigidly connected to a bogie frame, that is in turn connected to a fixed car body by linear suspensions: a couple of lateral springs and dampers and one torsional spring. The parameters used for the model as well as its Newton-Euler governing equations can be found in [Bigoni et al., 1]. The model has been developed in Python and solved by time integration routines belonging to the package SciPy[1].

The *half-wagon with Cooperrider bogie* vehicle is an extension of the simple Cooperrider bogie vehicle, where the car body is fixed but for the roll motion,

[1] http://www.scipy.org/



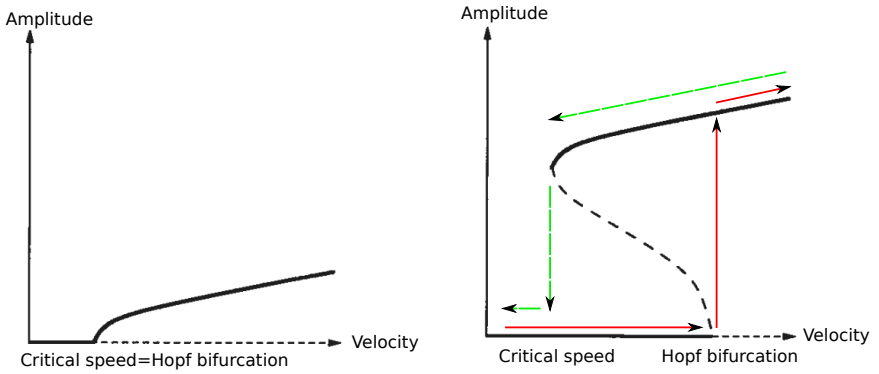
(a) Front view (b) Top view

Figure 9.2: The half/complete-wagon with Cooperrider bogie vehicle.

linear stiffness is added in the primary suspensions and both linear stiffness and damping are added in the secondary suspensions. Figure 9.2 shows the deployment of the suspensions and the relative positions of the vehicle's bodies. The vehicle is equipped with wheel-sets with the realistic wheel profile S1002 and runs on a track with cant 1/20 and rail profile UIC60. The parameters used for this vehicle can be found in [4, 5, 170].

The *complete-wagon with Cooperrider bogie* vehicle corresponds to the half-wagon with Cooperrider bogie vehicle, but for the fact that the car body is not fixed. The characteristics of this vehicle can be found in [6, 170].

The half-wagon and the complete-wagon with Cooperrider bogie vehicles are modeled using the Newton-Euler governing equations which are automatically obtained through the multi-body dynamics software DYnamics Train SImulation (DYTSI). The description of the software along with the study of its applications is available in [170]. The wheel-rail interaction is modeled using tabulated values generated with the routine RSGEO [171] for the static penetration at the contact points. These values are then updated using Kalker's work [172] for the additional penetrations. The creep forces are approximated using Shen-Hedrick-Elkins nonlinear theory [173]. The system of differential equations deriving from these modeling choices of the wheel-rail interaction is not only non-linear, but also non-smooth, due to the possible jumps in position of the contact patch.



(a) Super-critical Hopf bifurcation (b) Sub-critical Hopf bifurcation

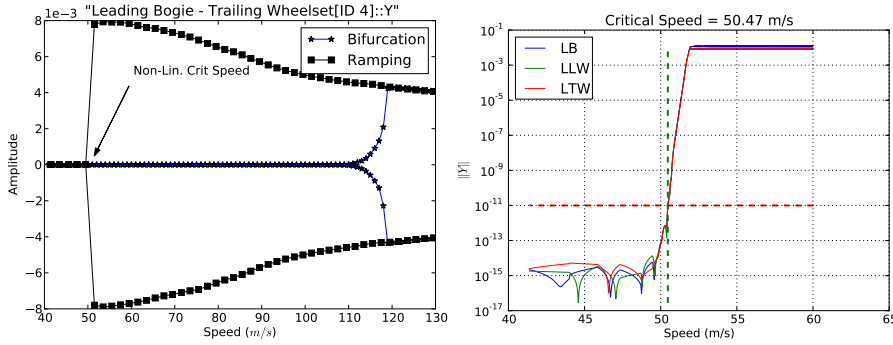
Figure 9.3: Sketches of a sub-critical and a super-critical Hopf bifurcation. Note that the super-critical Hopf bifurcation is the result of the collapse of the bifurcation point and the folding of the sub-critical Hopf bifurcation.

9.2 Identification of the QoI: the critical speed.

All the mathematical models used for the description of the dynamics of the vehicles presented here are characterized by strong non-linearities in the wheel-rail interaction forces. Above a design dependent *critical speed*, these non-linearities determine the appearance of sideways oscillations known as *hunting motion* [168]. The maximum allowed speed for vehicles is then determined in relation to the critical speed and the best designs available today are able to push this limit well above $300\,\mathrm{km\,h^{-1}}$ on straight tracks.

The hunting motion is due to the non-linearities appearing in the modeling of the wheel-rail contact, and the sideways oscillations appearing above the critical speed find explanation through the theory of non-linear dynamics and chaos [174]. The hunting motion appears as the result of an Hopf bifurcation which can be super or sub-critical [175]. Figure 9.3 shows two sketches of these bifurcations. At low speeds, only the central solution is stable and the system restores it whenever perturbed. When the speed is higher than the bifurcation point, then sideways oscillation appears and the system is attracted by the stable limit cycle. In the sub-critical case, this stable limit cycle extends below the bifurcation point, thus in the range of speeds between the critical speed and the bifurcation point there are two possible stable solution.

Hopf bifurcations are not the only non-linear effects that can appear in railway dynamics. For non-smooth systems chaos [7, 176, 177] has been observed as well. However, the vehicle models considered in our works related to UQ will



(a) Bifurcation diagram (b) Critical speed detection criterion

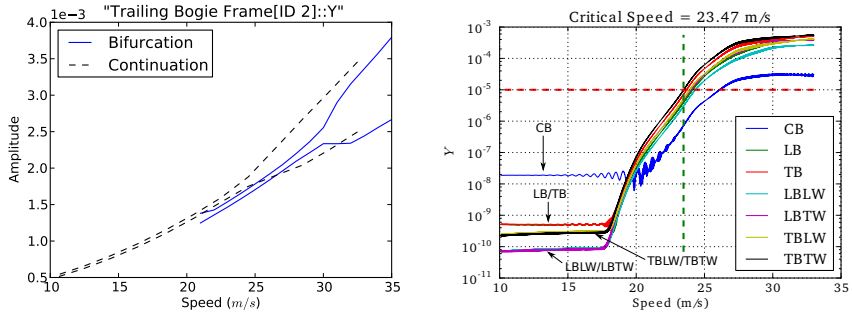
Figure 9.4: Bifurcation diagram and detection criterion for a vehicle running on straight track. Left: complete bifurcation diagram where the folding point is detected by continuation (ramping) method from the periodic limit cycle. Right: criterion for the determination of the critical speed based on the power of the lateral oscillations in a sliding window. LB, LLW and LTW stand for the bogie frame, the leading wheel set and the trailing wheel set respectively.

only exhibit an Hopf bifurcation in the range of speeds considered. We refer the reader to [7] for a review on non-smooth railway vehicle dynamics.

The right way to compute the critical speed is described in [178]. On straight track, a linearization of the system around the stable solution positioned at the center-line of the track can be used for the detection of the Hopf bifurcation. Then the stable limit cycle needs to be detected via numerical simulation and followed backwards until the fold of the sub-critical Hopf bifurcation disappears. This procedure for the detection of the fold of the sub-critical Hopf bifurcation is called *continuation* or *ramping*. On curved tracks the center-line of the track is not a stable solution, thus the bifurcation point needs to be detected through simulation as well.

The *gambling* way to detect the critical speed [178] consists in trying to enter the hunting limit cycle at a lower speed than the bifurcation point. Since one is interested only on the speed at which the folding disappear, this approach saves the time which would be wasted in the computation of an unnecessary part of the folding. However, in order to enter the hunting limit cycle, one needs to provide the right initial condition to the system. In the context of UQ where many simulations are often necessary, a candidate initial condition can be precomputed finding the full solution along the limit cycle. Since this limit cycle is an attractor, small perturbations within its domain of attraction will not drive the solution away from it.

In the following we will consider the critical speed as the QoI in several UQ analysis. Thus, it is useful to define a criterion for the automatic detection of the critical speed from numerical simulations. This criterion is based on the power of the lateral oscillations in a 1 s sliding window of the computed solution. In particular, a threshold is selected and the critical speed is defined as the speed at which the powers of the lateral displacement of all the components fall below such threshold. Figure 9.4 shows the bifurcation diagram of the half-wagon with Cooperrider bogie running on straight track along with the detection criterion of the critical speed. The threshold was conservatively set to 10^{-11}. Figure 9.5 shows the same analysis for the complete-wagon with Cooperrider bogie running on a curved track. Here the threshold was relaxed to 10^{-5}, in order to neglect unavoidable oscillations due to the slowly decreasing speed used along the curve.



(a) Bifurcation diagram (b) Critical speed detection criterion

Figure 9.5: Bifurcation diagram and detection criterion for a vehicle running on straight track. Left: complete bifurcation diagram where the folding point is detected by continuation (ramping) method from the periodic limit cycle. Right: criterion for the determination of the critical speed based on the power of the lateral oscillations in a sliding window. LB, LLW and LTW stand for the bogie frame, the leading wheel set and the trailing wheel set respectively.

The detection of the hunting motion, not only needs careful modeling, but also the right selection of the numerical integrator. In [7] we investigate the adoption of explicit and implicit numerical methods, leaning in the end toward the explicit ones which do not introduce unnecessary and sometimes misleading numerical damping. Furthermore, the choice is driven by the computational efficiency of explicit methods in the detection of the hunting motion.

Despite the selection of a good numerical integrator, the problem of accurately detecting the hunting motion is computationally demanding and the accuracy of the following UQ analysis will be mainly limited by this factor.

9.3 Sources of uncertainty: suspension components.

In all the analysis performed on the three models, the suspension components are considered to be subject to manufacturing uncertainties. Due to the lack of data, this uncertainty has been assumed to be Gaussian with 5% standard deviation around the suspension's nominal value. This assumption does not undermine the applicability of the method to other settings, where other distributions might be more suitable.

Uncertainties stemming from manufacturing errors can be considered to be independent. The characterization of these uncertainties should be carried out through extensive measurements.

The suspension components are also subject to uncertainties due to prolonged wear. It is likely that this kind of uncertainty have a clear structure due to the particular design of the vehicle and that uncertainty on the components is mutually dependent. Due to the lack of real data, we don't investigate this fact in this work.

9.4 Propagation of the uncertainty.

The propagation to the critical speed of the uncertainty stemming from manufacturing errors of the suspension components was first investigated in [179] using the MC method with good results.

In [180] PC methods – see section 5.2 –, in the form of gPC [46] and MEgPC [88, 89], were first applied on the dynamics of a two-degree-of-freedom quarter-car model.

In [Bigoni et al., 1–3] we apply gPC on the simple Cooperrider bogie vehicle. We compare both the Galerkin and the collocation methods to the MC and the QMC method in estimating the uncertainty of both the time-dependent evolution of the system at fixed speed, and of the critical speed of the model.

Figure 9.6 shows the distribution of the critical speed and the convergence of the three methods used on the simple Cooperrider bogie vehicle subject to uncertainties on the three suspension components – the model is considered completely symmetric here, so that the left and right stiffness and damping are changed together. We refer the reader to [Bigoni et al., 1–3] for more detailed results.

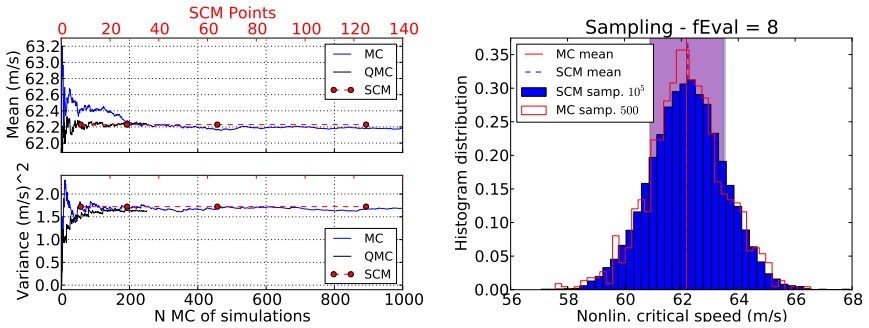


Figure 9.6: The PC collocation method – named Stochastic Collocation Method (SCM) – is compared to the MC and QMC methods in the estimation of the mean and variance (left) of the critical speed and in the approximation of its distribution (right). Note that on the convergence figure (left), the scale of the SCM method is positioned on the top axis.

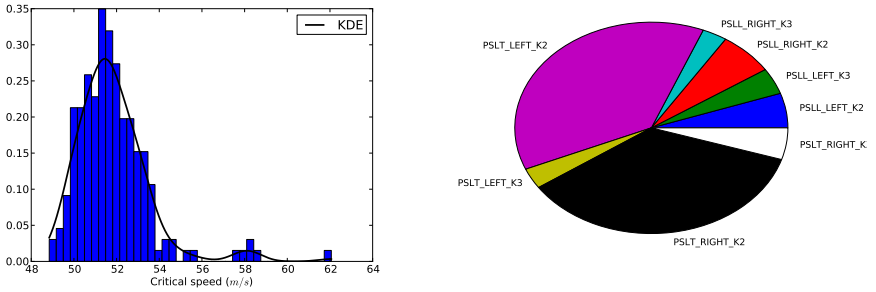


Figure 9.7: Sensitivity analysis on straight track. Left: histogram of the critical speed obtained using the LHC method and estimated density function obtained with the KDE method. Right: pie plot of the TSIs of the reduced stochastic model, where only the most influential components are analyzed. See table 9.1 for the definition of the notation used for the suspension components.

9.5 Sensitivity analysis.

The sensitivity of the critical speed to the uncertainty on the suspension components is presented in [Bigoni et al., 4–6]. In these works the left and right suspension components are considered separately, accounting for asymmetric vehicles which are normally appearing in reality due to uncertainties.

Straight track. In [Bigoni et al., 4, 5] the sensitivity of the critical speed on all the suspension components of the half-wagon with Cooperrider bogie vehicle is analyzed for the train running on straight track. The co-dimension[2] of the system is 24. The Total Sensitivity Indices (TSIs) are obtained using the ANOVA-HDMR estimated through the PC based Cut-HDMR – see section 5.3 and chapter 6. Different orders of approximation are used in order to construct a refined model with co-dimension 8, obtained neglecting uninfluential uncertainties due to the remaining 16 suspension components.

Figure 9.7 shows the propagation of the uncertainty obtained using the LHC method and the pie plot of the TSIs obtained for the refined model. Table 9.1 lists the numerical values of the TSIs of the components in the refined model along with the nominal value of the stiffness/damping parameters associated with each component. We refer the reader to [Bigoni et al., 4, 5] for detailed results.

For the studied setting of uncertainties, it was found that the most relevant components driving the critical speed on straight track are the primary longitudinal suspension components. Surprisingly the yaw dampers, which are known to be important in counteracting the hunting motion, showed to have little effect on the critical speed of the vehicle. In [Bigoni et al., 4] we provide some insight on this fact. UQ analysis are always based on some characterization of the input uncertainties. For the chosen uncertainties in this work, most of the probability mass of the distribution of the yaw damper is concentrated in a flat location of the response surface of the critical speed. Nonetheless, a different distribution could have had non-negligible mass concentrated in a more sensitive location of the response surface, leading to higher sensitivities for the yaw dampers.

Curved track. In [Bigoni et al., 6] the sensitivity on all the suspension components of the complete-wagon with Cooperrider bogie vehicle is analyzed for the train running on a curved track with curve radius 1600 m and super-elevation 110 mm. The co-dimension of the system is 48 and the same combination of techniques used on the straight track case is used for the study of the sensitivities. Figure 9.8 and table 9.2 show the results of the sensitivity analysis. The reader is referred to [Bigoni et al., 6] for detailed results. On the considered curved track and for the considered characterization of the sources of uncertainty, the critical speed was found to be greatly influenced by the yaw dampers in the leading secondary suspensions. This result is consistent with experience, where yaw dampers are known to play an important role in both the steering of the car in the curve and the stabilization of the dynamics of rail cars.

[2]The co-dimension is the dimension of the parameter space, which in this case is the number of suspension components in the system

Suspension	Nom. Value	TSI
PSLL_LEFT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.09
PSLL_LEFT_K3	$3646.0\,\mathrm{kN\,m^{-1}}$	0.07
PSLL_RIGHT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.11
PSLL_RIGHT_K3	$3646.0\,\mathrm{kN\,m^{-1}}$	0.05
PSLT_LEFT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.63
PSLT_LEFT_K3	$3646.0\,\mathrm{kN\,m^{-1}}$	0.05
PSLT_RIGHT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.59
PSLT_RIGHT_K3	$3646.0\,\mathrm{kN\,m^{-1}}$	0.08

Table 9.1: Sensitivity analysis on straight track. Nominal values of the suspension components and TSIs of the critical speed for the refined model. The naming convention used for the suspensions works as follows. PSL and SSL stand for primary and secondary suspension of the leading bogie respectively. The following L and T in the primary suspension stand for leading and trailing wheel sets. The last part of the nomenclature refers to the particular suspension components as shown in figure 9.2.

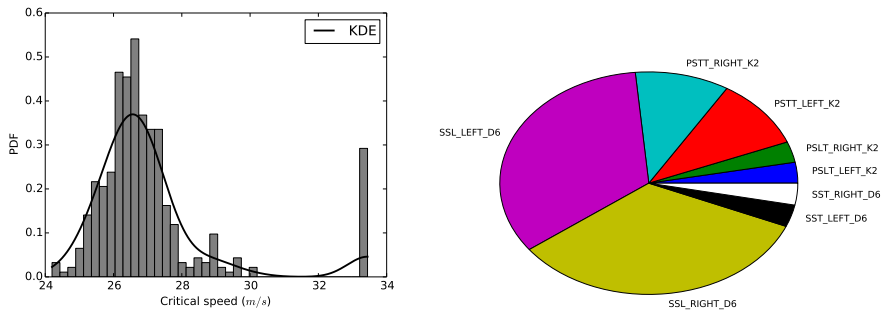


Figure 9.8: Sensitivity analysis on curved track. Left: histogram of the critical speed obtained using the LHC method and estimated density function obtained with the KDE method. Right: pie plot of the TSIs of the reduced stochastic model, where only the most influential components are analyzed. See table 9.2 for the notation used for the suspension components.

Suspension	Nom. Value	TSI
PSLT_LEFT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.09
PSLT_RIGHT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.09
PSTT_LEFT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.24
PSTT_RIGHT_K2	$3646.0\,\mathrm{kN\,m^{-1}}$	0.24
SSL_LEFT_D6	$166.67\,\mathrm{kN\,s/m}$	1.07
SSL_RIGHT_D6	$166.67\,\mathrm{kN\,s/m}$	1.07
SST_LEFT_D6	$166.67\,\mathrm{kN\,s/m}$	0.15
SST_RIGHT_D6	$166.67\,\mathrm{kN\,s/m}$	0.15

Table 9.2: Sensitivity analysis on curved track. Nominal values of the suspension components and TSIs of the critical speed for the refined model. The naming convention used for the suspensions works as follows. PS and SS stand for primary and secondary suspensions. The following L or T stands for leading or trailing bogie frame. The second L or T in the primary suspension stands for leading or trailing wheel sets. The last part of the nomenclature refers to the particular suspension components as shown in figure 9.2.

9.6 Conclusions and outlook

We have proven the applicability of UQ methods, based on PC, on a range of problems of increasing complexity in the field of railway vehicle dynamics. In the computationally expensive task of analyzing the uncertainty of the critical speed of railway vehicles, the UQ analysis benefits greatly from the fast convergence of PC methods.

This gain has been first showed on a low dimensional simple problem in section 9.4, where both the Galerkin and the collocation PC methods were tested in the propagation of the uncertainty due the manufacturing errors in the suspension components. Both of the methods showed a great improvement in the convergence rate with respect to random sampling methods. The Galerkin method was excluded in subsequent applications with more complex models, where its derivation was cumbersome and not practical due to a number of non-linear terms entering the system of equations.

The global sensitivity of the critical speed of two railway cars on straight and curved tracks to their suspension components has been analyzed in section 9.5 using the method of Sobol' based on the PC-HDMR approach. The suspension components are assumed to be subject to manufacturing errors with a certain distribution, and the methods highlights which of these components give the biggest contribution to the variance of the critical speed.

The method can be easily applied to other quantities of interest, and a logical

step would be its application to the characterization of the sensitivity of the lateral and vertical forces applied to the track by a railway vehicle running through a curve.

In order to move this application towards a real industrial setting, a complete characterization of the input uncertainties must be performed, starting from the collection of data regarding manufacturing uncertainties.

An even more important uncertainty suffered by the suspension components is caused by the wear. This uncertainty is added to the uncertainty due to manufacturing errors and leads to time-varying distributions of the suspension components. The random variables modeling the uncertainty in the suspension components will very probably be dependent, thanks to the coupling of the dynamics of the system, and this will most likely lead to challenging problems in the characterization of their distribution.

Chapter 10

Stochastic water wave simulation

Coastal, off-shore and maritime engineering are all subject to a number of uncertainties related to the weather conditions and the sea bed characteristics. The weather conditions are grouped into a number of sea states and each of these is characterized by a set of parameters, which determine particular waves, wind, currents, sea level and ice characteristics. The transition of sea states is usually modeled by stationary processes of the parameters of probability distributions describing the waves, wind, currents, sea level and ice conditions. Since safety, costs and profits in the coastal, off-shore and maritime industry are sensitive to these uncertainties, extensive measurements of environmental uncertainties have been and are being carried out [181–183].

In our contribution [Bigoni et al., 8] we focus on the propagation of uncertainties through a fully non-linear and dispersive model of water waves. The numerical solution of such a model is known to be computationally expensive and less accurate low-order models have been often preferred when applicable. However, recent developments in computer architectures and high-performance computing, thruogh the introduction of massively parallel algorithms, are allowing the solution of this high-order model in computationally feasable time [184]. Despite these developments, the simulation of water waves thorugh this model is still computationally expensive, and thus the UQ analysis of it can greatly benefit from the use of high-order methods which require a smaller number of function evaluations than traditional random sampling methods.

To highlight the potential of UQ analysis, we construct and propose new stochastic benchmarks based on traditional deterministic ones for testing UQ techniques, useful for proceeding towards advanced applications. The MC method and the high-order PC method in its collocation form are then tested on these benchmark problems. The results are compared to laboratory experimental results[1].

Several works have appeared in recent years treating uncertainties with PC methods in the computationally cheaper Shallow Water Equations (SWE). In [185] the MC and the PC method in the Galerkin and collocation form were compared when applied on the SWE modeling the propagation of a wave over a submerged hump. In [186] random sampling methods, sparse grids, PC in Galerkin and collocation form, and a novel quadrature technique called Compound Uncorrelated Dimension (CUD) quadrature were compared when applied on the SWE modeling flood prediction under an uncertain river bed topography and characteristics. In [187] a combination of non-intrusive (collocation) PC and ANOVA decomposition was used for the propagation and sensitivity analysis of the uncertain parameters entering the SWE modeling the runup of waves.

The propagation of uncertainties in coastal and off-shore engineering is often related to the ability of structures or ships to withstand fatigue due to external loadings [188, 189]. In many of these applications, the interest is focused on the prediction of extreme ocean environments [190], which can lead to catastrophic consequences. This topic is out of the scope of our contribution [Bigoni et al., 8] but it is a subject of ongoing research.

10.1 The mathematical and numerical models

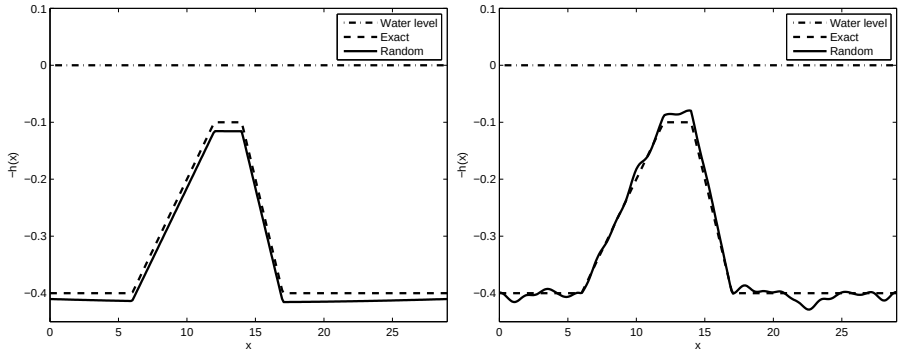
We consider unsteady water waves described by a potential model for two and three-dimensional fully nonlinear and dispersive free surface flows under the influence of gravity [191]. The flow is assumed inviscid and irrotational. The model can, without simplifications, be used for short and long wave propagation in both shallow and deep water where viscous and rotational effects are negligible. The sea bed is assumed variable and impermeable. The derivation of the model is presented in [184].

The model is characterized by non-linearities up to the fourth order, making its solution computationally challenging. Flexible-order finite differences are used to discretize the model [192], obtaining the numerical solver presented in [193].

[1]The benchmarks settings and data are made available at http://www2.compute.dtu.dk/ apek/OceanWave3D/

Description	Variable	Value
Bar height from bottom	h_{bar}	0.3 m
Bottom floor	h_b	−0.4 m
Entering wave period	T	2.02 s
Basin Length		29.0 m

Table 10.1: Nominal values and experimental settings used for the deterministic solution of the harmonic generation over a submerged bar.



(a) Correlation length $a = 1.0$ (b) Correlation length $a = 0.1$

Figure 10.1: Deterministic bottom topography and realizations of uncertain bottom topographies, modeled by the Gaussian random field with Ornstein-Uhlenbeck covariance used for the benchmark (T2D-4).

The chosen discretization is prone to the massive parallelization, which is treated in [184, 194, 195] and implemented in the multi-GPU solver OceanWave3D[2].

10.2 Deterministic and stochastic benchmarks

We consider two deterministic benchmarks which we then transform to stochastic benchmarks by assuming uncertainties on some of their parameters. The purpose of this exercise is to highlight opportunities for improved engineering analysis.

Harmonic generation over a submerged bar (2D) The first benchmark considered is the two dimensional experimental setting proposed by Beji et al.

[2]More details are available at http://www2.compute.dtu.dk/∼apek/OceanWave3D/

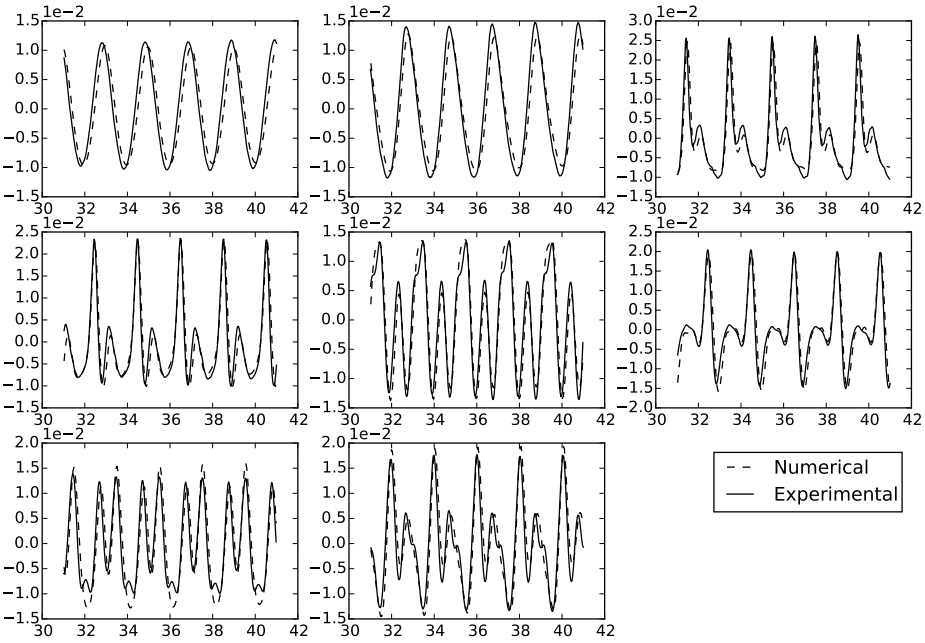


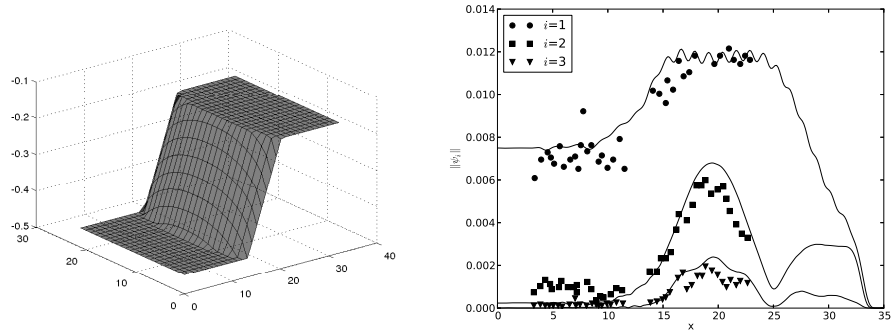
Figure 10.2: Deterministic solution of the submerged bar experiment at eight different gauge locations. The experimental data are due to Luth et al. [197].

[196] of the propagation of non-linear waves over a submerged bar. An experimental tank is considered, waves are generated at one end of the experimental domain and propagated through the tank. Figure 10.1 shows the deterministic bottom topography of this benchmark along with examples of uncertain bottom topographies. Table 10.1 lists the nominal values of the experiment.

The submerged bar causes a transformation of the wave due to the transfer of energy between its harmonics [198]. It is generally accepted that the experiment can be reproduced within engineering accuracy by the deterministic wave model considered here, which describe both the nonlinear and dispersive effects accurately.

Experimental measurements for this benchmark were carried out by Luth et al. [197] and are used here for comparison with the numerical results. Eight gauges are positioned at locations $x = \{4.0, 10.5, 13.5, 14.5, 15.7, 17.3, 19.0, 21.0\}$ and they measure the time-dependent wave amplitude. Figure 10.2 compares the experimental measurements with the numerical solutions obtained setting the parameters of the model to the nominal values used for the experiment.

The following stochastic benchmarks are constructed by the assumption of un-



(a) Bottom topography (b) Deterministic solution

Figure 10.3: Bottom topography and deterministic solution of the wave propagation in three dimensions. The first three harmonics of the numerical solution (full lines) at the center-line of the experimental domain are compared with the corresponding experimental measurements at different longitudinal locations in the basin (dots).

certainties on some of the model parameters:

(T2D-1) still water height[3]: $-h_b \sim \mathrm{tr}\mathcal{N}\left(0.4\,\mathrm{m}, 0.0125\,\mathrm{m}^2, [0.375\,\mathrm{m}, 0.425\,\mathrm{m}]\right)$,

(T2D-2) input wave period: $T \sim \mathcal{N}\left(2.02\,\mathrm{s}, 0.01\,\mathrm{s}^2\right)$

(T2D-3) still water height and input wave period: (T2D-1) and (T2D-2)

(T2D-4) bottom topography: $h \sim \mathcal{N}\left(\bar{h}, \sigma^2 C\right)$, where $\bar{h}$ is the nominal bottom topography, $\sigma^2 = 0.01^2$ and C is the Ornstein-Uhlenbeck covariance (B.36) with $a = 1.0$. One realization of this setting is also show in figure 10.1a. This model tries to capture small macroscopic uncertainties in the slope of the basin's bottom.

Harmonic generation over a semi-circular shoal (3D) The second benchmark is the three dimensional propagation of a regular wave over a semi-circular shoal, based on the experiments in [199]. Figure 10.3a shows the bottom topography of the experiment and table 10.2 lists the nominal values of the input wave characteristics. The result of the experiment, with the prescribed nominal values of its parameters, is decomposed into its harmonics. In figure 10.3b the evolution of the first three harmonics along the experimental domain is compared to the measured harmonics.

The following stochastic benchmarks are constructed by the assumption of uncertainties on some of the model parameters:

[3] $\mathrm{tr}\mathcal{N}$ stands for the truncated normal distribution.

Description	Variable	Value
Entering wave height	H	$0.015\,\mathrm{m}$
Entering wave period	T	$2.0\,\mathrm{s}$

Table 10.2: Nominal values and experimental settings used for the deterministic solution of the harmonic generation over a semi-circular shoal.

(T3D-1) input wave height: $H \sim \mathcal{N}\left(0.015\,\mathrm{m}, 0.75 \times 10^{-6}\,\mathrm{m}^2\right)$

(T3D-2) input wave period: $T \sim \mathcal{N}\left(2.0\,\mathrm{s}, 0.01\,\mathrm{s}^2\right)$

(T3D-3) input wave height and period: (T3D-1) and (T3D-2)

10.3 Uncertainty quantification

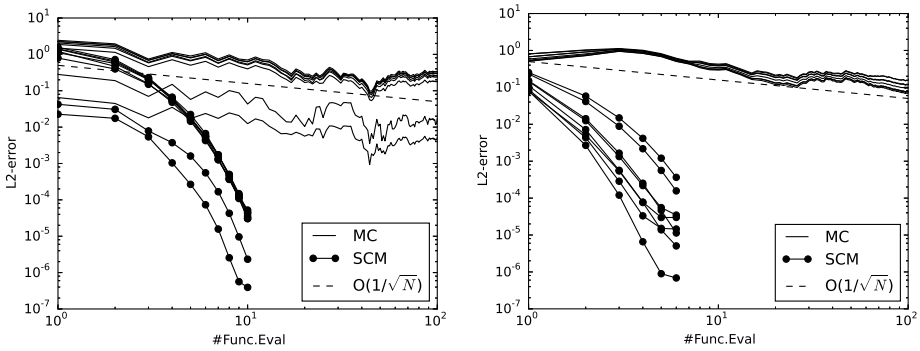
Even if the constructed benchmarks do not aim at representing the real uncertainty in the practical experiments, they are compared to laboratory experiments and thus all the uncertainties can be considered to be epistemic.

The methods applied in this work are the MC method and the collocation PC method, known also as the Stochastic Collocation Method (SCM). Collocation PC has been applied both in its tensorized form and in the non-adaptive Sparse Grid form. MC method has been applied to all the experimental settings for validation purposes. However, for the low-dimensional problems considered, it was always outperformed by PC based methods, so the results showed in [Bigoni et al., 8] are always the ones obtained using PC methods.

In the following we will present the main results obtained. The interested reader is referred to [Bigoni et al., 8] for more detailed results.

Harmonic generation over a submerged bar (2D) Figure 10.4 shows the convergence rates of MC and SCM for the benchmarks (T2D-1) and (T2D-2). We can see that the SCM method converges exponentially fast to the solution whereas the MC converges with its typical $\mathcal{O}(1/\sqrt{N})$ rate. The convergence of the SCM method correctly flattens around 10^{-6}, which is the accuracy limit set in the deterministic solver.

Figure 10.5 shows the probability distribution of $10\,\mathrm{s}$ of the steady state of the solution of benchmark (T2D-3), along with its mean and 95% confidence interval obtained using the SCM. In [Bigoni et al., 8] we show that the two uncertainties entering the benchmark (T2D-3) have a non-trivial combined influence on the resulting wave dynamics, which could not be explained by simple superposition.



(a) Uncertain still water height (b) Uncertain wave length

Figure 10.4: Convergence rate of the MC method and the SCM method on the submerged bar benchmarks (T2D-1) and (T2D-2). The L^2 error of the approximation of 10s of simulation is computed against an highly accurate reference solution of order 20. The different lines belong to different gauges. The MC method exhibit its slow convergence of $\mathcal{O}(1/\sqrt{N})$, while the SCM method shows *spectral* convergence.

Note also how little informative the 95% confidence interval is of the uncertainty in the solution at some of the gauges locations, where the distribution is clearly non-Gaussian. While the fixed-time probability distribution upstream of the submerged bar resemble Gaussian distributions, the probability distributions downstream of the bar are often multimodal due to the phase shifting effect that the two input uncertainties have.

Figure 10.6 shows the application of non-adaptive Sparse Grids of sufficient level $l = 3$ on benchmark (T2D-4), where the random field is expanded using a KL-expansion retaining 95% of the total variance, resulting in a truncated KL-expansion with 3 terms. We can see that the uncertain bottom topography considered plays an important role in the wave transformation downstream of the bar, even if the random field considered has a relatively long correlation length and small variance.

Harmonic generation over a semi-circular shoal (3D) Here we show the results regarding benchmark (T3D-3). Figure 10.7a shows the space-dependent probability distribution of the first three harmonics of the solution along with the mean, the variance, the 95% confidence interval and the experimental results. We can clearly spot how the change of the slope in the bottom topography – see figure 10.3a – not only influences the deterministic solution but also the transformation of its probability distribution.

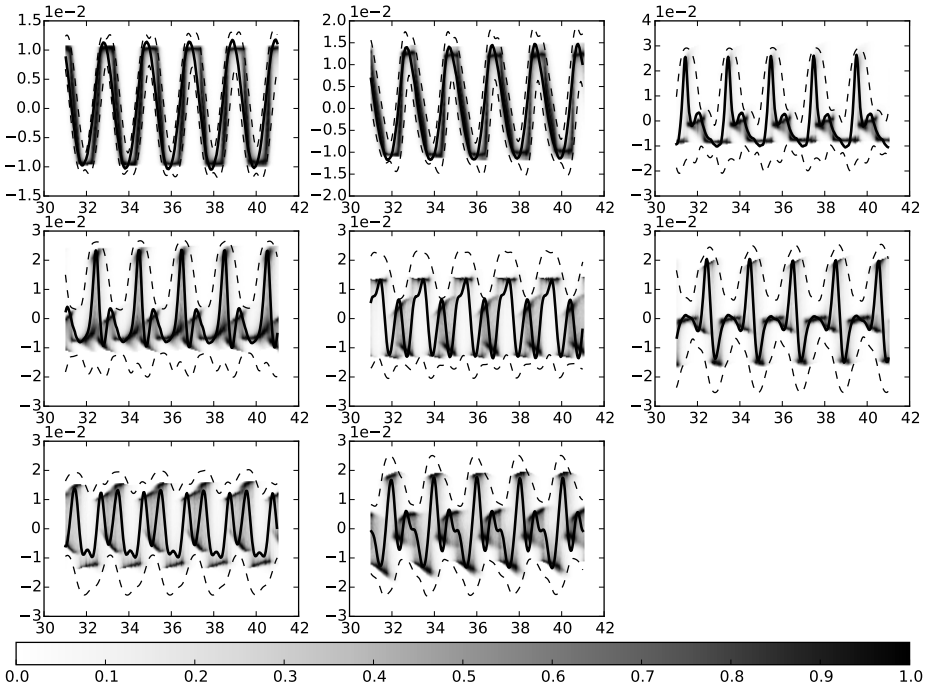


Figure 10.5: Probability distributions of the time-varying solution of the submerged bar benchmark (T2D-3) with uncertain still water height h_b and wave period T, at different measurement locations. The thick black lines show the experimental results at the different gauges, while the dashed lines show the 95% confidence intervals.

A 5-th order PC expansion was used for this test, resulting in 36 function evaluations. By a close investigation of the decay of the generalized Fourier coefficients in figure 10.7b, we can deduce that the PC approximation struggles to converge in the region $x \in [20, 30]$, where we see the solution drifting away from the usual Gaussian shape. This is clearly due to the slope in the bottom topography. Furthermore, note that the generalized Fourier coefficients in figure 10.7b are sorted with increasing orders, listing first the coefficients in the direction of the wave period[4]. Thus, the decay rate shown in figure 10.7b suggests that the PC expansion struggles more in the approximation in the direction of the wave period, where the decay is slower, than on the direction of the wave height.

[4]This means that if $\{c_{i,j}\}_{i,j=0}^{5}$ are the generalized Fourier coefficients, with i listing the basis for the uncertain wave height and j listing the basis for the uncertain wave period, the listing in figure 10.7b goes as follows: $c_{0,0}, c_{0,1}, c_{0,2}, \ldots, c_{1,0}, \ldots$.

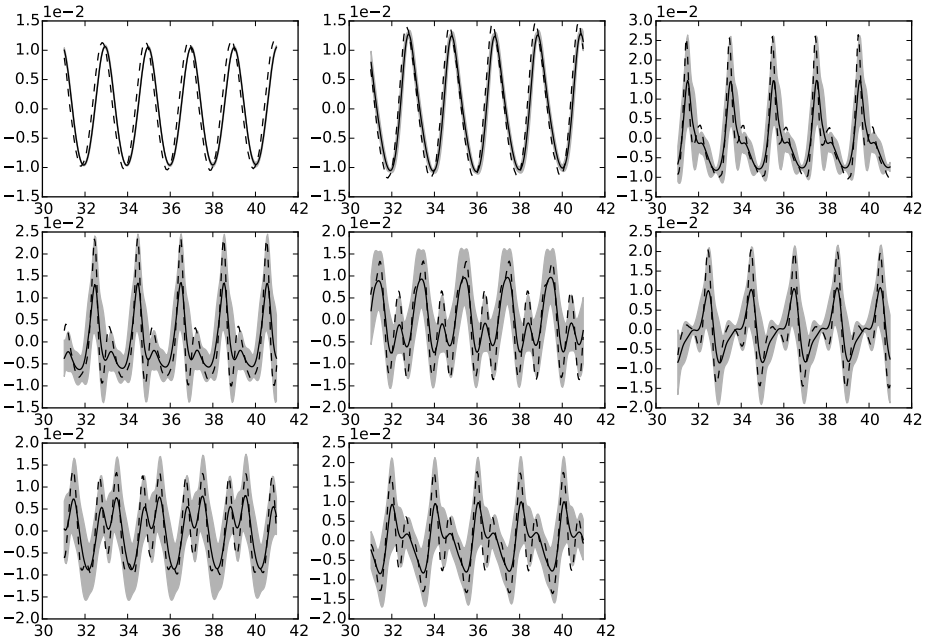
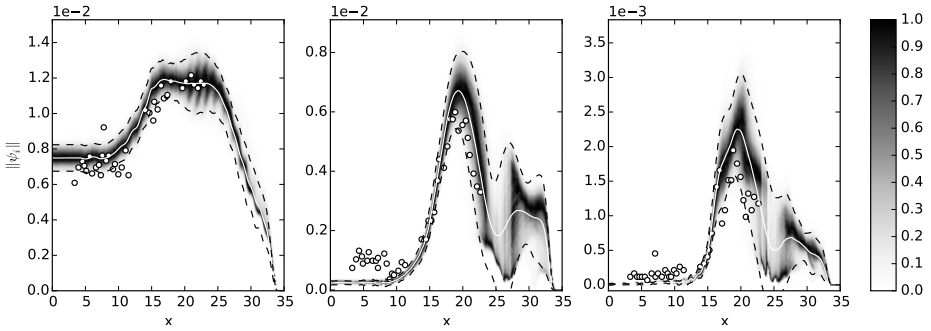


Figure 10.6: Mean (solid line) and standard deviation (shaded) of the solution of the submerged bar benchmark (T2D-4) at the different measurement locations. The experimental data (dashed line) is also shown. The results are obtained using the Sparse Grid method with 19 function evaluations and checked against the MC method.

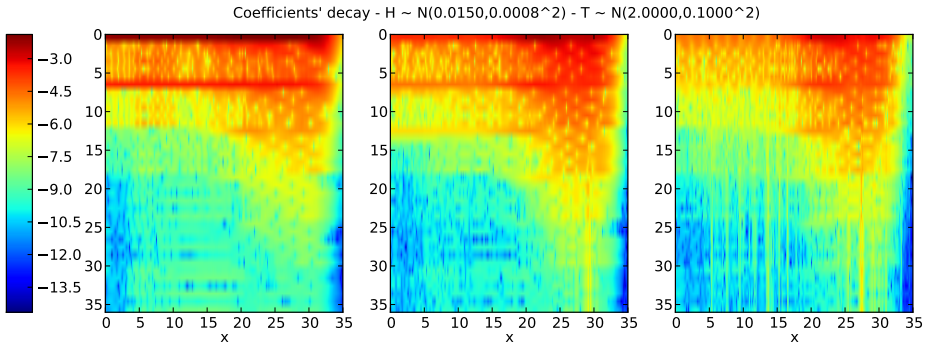
10.4 Conclusions and outlook

Two and three dimensional stochastic benchmarks for the propagation of waves under uncertain input conditions have been constructed and used to demonstrate the applicability of PC based methods for the propagation of such uncertainties. The collocation approach has been preferred over the Galerkin approach due to its applicability to already developed state-of-the-art software for the solution of a fully non-linear and dispersive potential flow model of water waves. Despite the high efficiency of the implementation, the UQ analysis of this kind of model is computationally demanding. Thus, the UQ analysis benefits greatly from the possibility given by collocation methods of using parallel resources with no additional implementation burden.

The non-linear nature of the problem and the varying bottom characterizing the considered benchmarks lead to complex transformations of the input Gaussian distributions to distributions which are often multi-modal. Nevertheless, the



(a) Space-dependent probability distribution



(b) Space-dependent decay of the generalized Fourier coefficients

Figure 10.7: Space-dependent probability distribution and decay of the generalized Fourier coefficients in benchmark (T3D-3). In the top figure, the white solid line represent the mean for the three harmonics. The dashed lines show the 95% confidence interval around the mean. The scattered dots are the experimental measurements. In the bottom figure the 36 generalized Fourier coefficients are sorted top-to-bottom from the lowest order to the highest, ranging first along the direction of the wave period.

PC method used is able to detect such transformations and approximate the distributions accurately. The work has thus highlighted the potential of the PC method in the context of an application with a small number of uncertainty parameters, but with a big computational burden.

Ongoing works are focusing on tackling problems with an higher number of uncertainties affecting the water wave dynamics and ultimately the extreme loads on off-shore structures. The uncertainties considered stem from the lack of accurate measurements of the bathymetry and its change over time due to sedimentation. Surrogate models such as the sparse grids pseudo-spectral method [49–51] and the spectral tensor-train decomposition [Bigoni et al., 9] are being used to accelerate this analysis.

Part III

Appendices

Appendix A

Dynamical systems

In this work we will consider dynamical systems in the form of *Differential Equations* (DE). These express the relation of an unknown function with its derivatives. Differential equations are used to describe physical phenomena defined on continuous domains (space, time, etc.). Our attention will be focused on real valued solutions of DE.

Definition A.1 (Ordinary Differential Equation - ODE)
Let $t \in T = [0, T_f] \subset \mathbb{R}$ and $u \in \mathcal{C}^n(T, \mathbb{R})$ be an unknown real-valued function satisfying

$$F\left(t, u, u', \ldots, u^{(n)}\right) = 0 \tag{A.1}$$

where $u^{(i)} = \mathrm{d}^{(i)}u / \mathrm{d}t^{(i)}$. Equation (A.1) is an ODE of order n.

The independent variable t represents usually the time domain. Some ODEs can be written in *normal form*:

$$u^{(n)} = f\left(t, u, u', \ldots, u^{(n-1)}\right) \tag{A.2}$$

and through the definition of $\mathbf{u} \in \mathcal{C}^n(T, \mathbb{R}^n)$ by

$$\mathbf{u}_1 = u, \quad \mathbf{u}_2 = u', \quad \ldots \quad \mathbf{u}_n = f\left(t, u, u', \ldots, u^{(n-1)}\right)$$

we can rewrite (A.2) into the system of first order ODEs

$$\mathbf{u}' = g(t, \mathbf{u}) = \left[u, u', \ldots, f\left(t, u, u', \ldots, u^{(n-1)}\right)\right]^T \tag{A.3}$$

In case t appears explicitly in the definition of f, the system will be called *non-autonomous*. Non-autonomous systems can be converted to equivalent *autonomous* ones by the insertion of the additional dummy variable $u_{n+1} = x$. The equation $u'_{n+1} = 1$ can then be included in the definition of the problem. The autonomous system will then be written as

$$\mathbf{u}' = g(\mathbf{u}) \tag{A.4}$$

Particular solutions passing by a specified point $(t_0, \mathbf{u}_0)$ can be determined for (A.4). This is an *Initial Value Problem*.

Definition A.2 (Initial Value Problem (IVP)) *Given an n-th order ODE, an Initial Value Problem is finding $\mathbf{u} \in \mathcal{C}^n(T, \mathbb{R}^n)$ such that*

$$\begin{cases} \mathbf{u}' = g(\mathbf{u}) \\ \mathbf{u}^{(k)}(t_0) = \mathbf{u}_0 \quad \text{for } k = 1, \ldots, n-1\ . \end{cases} \tag{A.5}$$

The values $(t_0, \mathbf{u}_0)$ are called initial values.

It can be shown that under continuity conditions known as Lipschitz, the IVP (A.5) has an unique local, and sometimes global, solution. We refer the reader to one of the many books on ODEs [20, 54] for further properties and solutions of such problems.

When the function u is multivariate, partial derivatives with respect to different dimensions can define Partial Differential Equations.

Definition A.3 (Partial Differential Equation - PDE)
Let $\mathbf{x} \in \mathbb{R}^d$ and let $u \in \mathcal{C}^n(\mathbb{R}^d, \mathbb{R})$ be a multivariate unknown real-valued function satisfying

$$F\left(x_1, \ldots, x_n, u, u_{x_1}, \ldots, u_{x_d}, \ldots, u_{x_1, x_d}, \ldots\right) = 0 \tag{A.6}$$

where $u_{x_i} = \frac{\partial u}{\partial x_i}$ and the maximum order of derivation is n. Equation (A.6) *is a PDE of order n.*

The condition $u \in \mathcal{C}^n(\mathbb{R}^d, \mathbb{R})$ can in general be relaxed because only the partial derivatives involved in (A.6) need to be defined and continuous. An equivalent notation of (A.6) is given by

$$\mathcal{L}\mathbf{u} = 0 \tag{A.7}$$

where $\mathcal{L} : \mathcal{C}^n(D, \mathbb{R}^m) \to \mathcal{C}(D, \mathbb{R}^m)$ is a – possibly non-linear – differential operator.

A time-dependent PDE can be viewed in the context of the same definition A.3 by setting for example $t = x_1 \in T = [0, T_f]$. For clarity in the following we will denote $D \subset \mathbf{R}^d$ the spatial domain of the problem, $\mathbf{x} \in D$ the spatial variable and for a time-dependent PDE, $u \in \mathcal{C}^n(T \times D, \mathbb{R})$. PDEs can be defined on vector valued functions that we will denote $\mathbf{u} \in \mathcal{C}^n(T \times D, \mathbb{R}^m)$. This form includes also the case in which the PDE can be written in an autonomous system of PDEs in normal form and reduced by the insertion of dummy variables – as it was done in (A.4) – obtaining[1]

$$\mathbf{u}_t = \mathcal{G}\mathbf{u} \tag{A.8}$$

This formulation is often useful for the solution of time-dependent PDEs by the Method of Lines, where the operator $\mathcal{G}$ is first discretized to obtain a set of semi-discrete ODEs which can be solved by one of the many time-stepping methods available [23].

Particular solutions of the PDE (A.6)-(A.7) can be found for particular domains D if particular values are defined for its boundaries ∂D.

Definition A.4 (Boundary Value Problem - BVP)
Let $D \subset \mathbb{R}^d$ be compact of dimension d and let ∂D be the $d-1$ dimensional manifold bounding D. A Boundary Value Problem is formed by the n-th order PDE and a set of conditions for the solution on the boundary

$$\begin{cases} \mathcal{L}\mathbf{u} = 0 & \mathbf{x} \in D \\ \mathcal{B}\mathbf{u} = 0 & \mathbf{x} \in \partial D \end{cases} \tag{A.9}$$

where $\mathcal{B}$ is a – possibly non-linear – boundary differential operator.

If these equations define a unique solution, then the problem is said to be *well posed*. Proving the well posedness of such a boundary value problem is a non-trivial task, and we refer the reader to the extensive existing literature on the topic [21, 200].

When a PDE is time dependent, in order to look to particular solutions, conditions must be specified both at the boundaries and at some fixed time, usually the initial. This results in an initial value problem for PDEs.

[1] This is to be intended for $\mathbf{u} \in \mathcal{C}^n(D, \mathbb{R}^{m \times (n+1)})$

Definition A.5 (Initial Value Problem (IVP)) *Given an n-th order PDE, the Initial Value Problem is to find $\mathbf{u} \in \mathcal{C}^n(T \times D, \mathbb{R}^m)$ such that*

$$\begin{cases} \mathbf{u}_t = \mathcal{G}\mathbf{u} & (t, \mathbf{x}) \in T \times D \\ \mathcal{B}\mathbf{u} = 0 & (t, \mathbf{x}) \in T \times \partial D \\ \mathbf{u} = \mathbf{u}_0 & (t, \mathbf{x}) \in T = t_0 \times D \end{cases} \tag{A.10}$$

Appendix B

Probability theory and functional spaces

In the following sections the measure theoretic approach to probability will be shortly presented in order to fix the notation used along this work. For a deeper introduction to measure theoretic probability theory the reader is referred to one of the many books on the topic [18, 19].

B.1 Probability space

Whether randomicity is a property of nature or a limit in human observation, the word random has pervaded the human vocabulary, sometimes inappropriately, in the last century[1]. At the heart of the term random there are *events* which we will group in the *space of events* Ω. Our ultimate goal is to assign a non-negative real number to subsets of Ω, which will represent its measure and can be thought as its volume or, from a statistician perspective, the odds that the particular event will occur. However, without other constraints, the space Ω can be rather complex and its power set 2^{Ω} can lack properties which would allow the definition of a "reasonable" measure on its elements[2]. Thus we will

[1] https://books.google.com/ngrams/graph?content=random

[2] See for example the "Vitali set" [18].

work with more manageable *σ-algebras* – also called σ-fields –, namely the sets $\mathscr{F} \subset 2^\Omega$ such that

- $\Omega \in \mathscr{F}$,
- $\mathscr{F}$ is closed under complementation ($F \in \mathscr{F} \Rightarrow F^C \in \mathscr{F}$),
- $\mathscr{F}$ is closed under countable unions ($F_1, F_2, \ldots \in \mathscr{F} \Rightarrow (\cup_{i=1}^{\infty} F_i) \in \mathscr{F}$).

A commonly used σ-algebra in probability theory is the *Borel* σ-algebra $\mathscr{B}(\mathbb{R}^k)$ – just denoted by $\mathscr{B}$ in the following – on $\mathbb{R}^k$, which is the σ-algebra generated by the open sets in $\mathbb{R}^k$. Being generated by the open sets in $\mathbb{R}^k$ means that $\mathscr{B}(\mathbb{R}^k)$ is the smallest σ-algebra containing the open sets.

σ-algebras allow the definition of *measures* $\pi : \mathscr{F} \to \overline{\mathbb{R}}$ satisfying the following properties

- non-negativity: $\pi(F) \geq 0, \quad \forall F \in \mathscr{F}$,
- zero measure of the empty set: $\pi(\emptyset) = 0$,
- countable additivity: For all countable collections $\{F_i\}_{i=0}^{\infty} \subset \mathscr{F}$ with disjoint elements, $\pi\left(\cup_{i=1}^{\infty} F_i\right) = \sum_{i=1}^{\infty} \pi(F_i)$.

A measure is called *finite* if $\pi(\Omega) < \infty$. In probability theory we mostly work with finite measures with $\pi(\Omega) = 1$, representing the fact that the set of all events has probability 1 to occur. To distinguish *probability measures* on Ω from other measures, they will be denoted by $P : \mathscr{F} \to [0, 1]$. The triple $(\Omega, \mathscr{F}, P)$ is called the *probability space*.

Probability theory make extensive use also of *σ-finite* measures. A measure is called σ-finite if the set Ω can be decomposed into a countable union of measurable sets – i.e. belonging to $\mathscr{F}$ – with finite measure. The most commonly used σ-finite measure is the the *Lebesgue measure* on the real line $\lambda : \mathscr{B}(\mathbb{R}) \to \overline{\mathbb{R}}$ defined by $(a, b) \mapsto (b - a)$. The Lebesgue measure can similarly be defined on $\mathscr{B}(\mathbb{R}^k)$.

The intuitive way of thinking about a measure is to consider the Lebesgue measure and realize that it assign to any set in $\mathscr{B}(\mathbb{R}^k)$ its volume. Singletons in $\mathbb{R}^k$ have Lebesgue measure zero. We will say that a property holds *almost everywhere* (a.e.) if the measure of the set where the property doesn't hold is zero. Sets of measure zero can be rather complex, for example consider the set $\mathbb{Q}$ which in spite of being dense in $\mathbb{R}$, it has $\lambda(\mathbb{Q}) = 0$.

B.2 Random variables

Random variables are the core objects that are investigated in probability theory. In spite of the name, random variables are functions.

Definition B.1 (Random variable) *Let $(\Omega, \mathscr{F}, P)$ be a probability space and let $(\Omega_1, \mathscr{F}_1)$ be another space with the associated σ-algebra. A random variable X is a function*

$$X : \Omega \to \Omega_1 \ , \tag{B.1}$$

which is $\mathscr{F} - \mathscr{F}_1$-measurable, i.e. such that

$$X^{-1}(F) \in \mathscr{F} \qquad \forall F \in \mathscr{F}_1 \ , \tag{B.2}$$

where X^{-1} denotes the pre-image of F.

For example, real valued continuous random variables are defined with $(\Omega_1, \mathscr{F}_1) = (\mathbb{R}, \mathscr{B})$. In the following we will work mainly with this kind of random variables, so the remaining of the presentation will be focused on them.

A random variable is characterized by its *probability distribution* which is expressed by the measure π as[3]

$$\pi(F) = P(X \in F) = P(\{\omega \in \Omega : X(\omega) \in F\}) = \int_F \pi(\,\mathrm{d}x) \ , \tag{B.3}$$

defined for any $F \in \mathscr{B}$ and where $\pi(\,\mathrm{d}x) = P(\omega \in X^{-1}(\,\mathrm{d}x))$[4]. The integral in (B.3) has to be interpreted in the sense of Lebesgue [18, 19]. We will use the notation $X \sim \pi$ to express the fact that X is a random variable with probability distribution π. Commonly used distributions are

- the *Normal/Gaussian distribution* $\mathcal{N}(\mu, \sigma^2)$, where μ is the mean and σ^2 is the variance – see sec. B.5,
- the *Beta distribution* $\mathcal{B}e(\alpha, \beta)$, where $\alpha, \beta > 0$,
- the *Uniform distribution* $\mathcal{U} = \mathcal{B}e(1, 1)$,
- the *Gamma distribution* $\Gamma(k, \theta)$, for $k, \theta > 0$.

It is common to characterize real valued random variables by their Cumulative Distribution Function (CDF)

$$F_X(x) = P(X \leq x) = \int_{-\infty}^{x} \pi(\,\mathrm{d}s) \ . \tag{B.4}$$

[3] The measurability of X plays an important role here.

[4] In literature the notation $\pi(\,\mathrm{d}x)$ is sometimes written $\mathrm{d}\pi(x)$ even if this is not perfectly consistent from the measure theoretic perspective.

If the measure π is *absolutely continuous* with respect to the Lebesgue measure λ, i.e. $\lambda(F) = 0$ implies $\pi(F) = 0$ for all $F \in \mathscr{B}$, by the *Radon-Nikodym theorem* there exists the density $\rho : \mathbb{R} \to \mathbb{R}^+$ such that

$$\pi(F) = \int_F \rho_X(x)\lambda(\,\mathrm{d}x) = \int_F \rho_X(x)\,\mathrm{d}x\ , \tag{B.5}$$

where in the last equality the notation was simplified for the Lebesgue measure. The function ρ is called the Probability Density Function (PDF) of X.

Another way of characterizing the distribution of X is through its characteristic function, but its treatment is out of the scope of this work, so the reader is referred to [18] for further reading.

We call $\mathbf{X} = (X_1, \ldots, X_n)$ a *random vector*, where X_i are one dimensional real-valued random variables. The random vector $\mathbf{X}$ is associated to a distribution π defined on $\mathscr{B}(\mathbb{R}^n)$. In some cases this distribution can be a *product measure* of the form $\pi = \prod_{i=1}^d \pi_i$, where $X_i \sim \pi_i$. In these cases we will say that the random variables are *independent*.

Since the distribution π assigns probabilities to the events in $\mathscr{B}(\mathbb{R}^n)$, it is reasonable to be able to *sample* $\mathbf{X}$ accordingly to π, obtaining the *realization* $\mathbf{X}(\omega)$. In many cases we would like to sample the random vector many times, then we will define the set of random vectors $\{\mathbf{X}^{(j)}\}_{j=1}^N$ from which to sample. We will usually require an additional condition on $\{\mathbf{X}^{(j)}\}_{j=1}^N$, namely that they must be *independent and identically distributed* (i.i.d.) random vectors. Then a particular *sample* or *realization* of $\{\mathbf{X}^{(j)}\}_{j=1}^N$ will be denoted by $\{\mathbf{X}^{(j)}(\omega)\}_{j=1}^N$. This is commonly called an *ensemble*.

B.3 The $L^p(\Omega, \mathscr{F}, P)$ and $L^p_\pi(\mathbb{R})$ spaces

Different spaces of real valued random variables can be defined depending on the order of their integrability. Given the probability space $(\Omega, \mathscr{F}, P)$ and for $1 \leq p < \infty$, the L^p space of $\mathbb{X}$-valued random variables is

$$L^p(\Omega, \mathscr{F}, P; \mathbb{X}) = \left\{ X : \Omega \to \mathbb{X} : \int_\Omega |X(\omega)|^p P(\,\mathrm{d}\omega) < \infty \right\}\ . \tag{B.6}$$

For simplicity we will omit the range of the random variables when $\mathbb{X} = \mathbb{R}$ and denote the L^p space by $L^p(\Omega, \mathscr{F}, P)$. The space $L^1(\Omega, \mathscr{F}, P)$ is called the space of *integrable* random variables. The $L^2(\Omega, \mathscr{F}, P)$ space is the space of variables with *finite variance*. For $p = \infty$, the $L^\infty(\Omega, \mathscr{F}, P)$ space of a.e. bounded random variables is defined by

$$L^\infty(\Omega, \mathscr{F}, P) = \left\{ X : \Omega \to \mathbb{R} : \operatorname*{ess\,sup}_{\omega \in \Omega} [X(\omega)] < \infty \right\}\ , \tag{B.7}$$

where $\operatorname{ess\,sup}_{\omega\in\Omega}[X(\omega)] = \inf\{a \in \mathbb{R} : P(\{\omega \in \Omega : X(\omega) > a\}) = 0\}$. The spaces $L^p(\Omega, \mathscr{F}, P)$ are all Banach spaces [55, 56] with norm defined by

$$\begin{aligned} \|X\|_{L^p(\Omega,\mathscr{F},P)} &= \left(\int_\Omega |X(\omega)|^p P(\,\mathrm{d}\omega)\right)^{\frac{1}{p}} \quad \text{for } 1 \le p < \infty\,, \\ \|X\|_{L^\infty(\Omega,\mathscr{F},P)} &= \operatorname*{ess\,sup}_{\omega\in\Omega}[X(\omega)] \qquad\qquad\qquad \text{for } p = \infty\,. \end{aligned} \tag{B.8}$$

The space $L^2(\Omega, \mathscr{F}, P)$ is a Hilbert space [55, 56] with inner product defined by

$$(X, Y)_{L^2(\Omega,\mathscr{F},P)} = \int_\Omega X(\omega)Y(\omega)P(\,\mathrm{d}\omega)\,. \tag{B.9}$$

In spite of being functions, random variables owe their names to the fact that they are commonly used as arguments of functions. The real valued function f applied to the real valued random variable X generates the random variable $Y = f(X) \sim f(\pi)$, where $f(\pi)$ is the transformed probability distribution of Y. The random variable Y is a new random variable defined on $(\Omega, \mathscr{F}, P)$, so the definition of the L^p spaces for the functions f is closely related to the ones already presented. For the probability space $(\Omega, \mathscr{F}, P)$, the random variable $X : \Omega \to S \subset \mathbb{R}$ with distribution π, and for $1 \le p < \infty$, the L^p space of functions of X is

$$\begin{aligned} L^p_\pi(S) &= \left\{ f : S \to \mathbb{R} : \int_S |f(x)|^p \pi(\,\mathrm{d}x) < \infty \right\}\,, \\ L^\infty_\pi(S) &= \left\{ f : S \to \mathbb{R} : \operatorname*{ess\,sup}_{x\in S}[f(x)] < \infty \right\}\,. \end{aligned} \tag{B.10}$$

where we remind that $\pi(\,\mathrm{d}x) = P(\omega \in X^{-1}(\,\mathrm{d}x))$. The space $L^1_\pi(S)$ is the space of *integrable* functions, the space $L^2_\pi(S)$ is the space of *square integrable* functions while the space $L^\infty_\pi(S)$ is the space of functions *bounded almost everywhere*. The space $L^2_\pi(S)$ is an Hilbert space while the spaces L^p are only Banach for any other p. The norms and the inner product are defined accordingly:

$$\begin{aligned} \|f\|_{L^p_\pi(S)} &= \left(\int_S |f(x)|^p \pi(\,\mathrm{d}x)\right)^{\frac{1}{p}}\,, \\ \|f\|_{L^\infty_\pi(S)} &= \operatorname*{ess\,sup}_{x\in S}[f(x)]\,, \end{aligned} \tag{B.11}$$

$$(f, g)_{L^2_\pi(S)} = \int_S f(x)g(x)\pi(\,\mathrm{d}x)\,. \tag{B.12}$$

B.4 The $\mathcal{C}^k(S)$ and the $\mathcal{H}^k_\pi(S)$ Sobolev spaces

Along this work we refer to the *regularity* of a function f as its maximum degree of differentiability. We distinguish between strong and weak differentiability by the definition of two different spaces of functions. We define the *class* $\mathcal{C}^k(S)$ to be

$$\mathcal{C}^k(S) = \left\{ f : S \to R \ : f^{(i)} \text{ exists and is continuous } \forall i \in [0,k] \right\} , \tag{B.13}$$

where $f^{(i)}$ is the *strong derivative* of f. This definition is generalized to the multidimensional case, with $f^{(i)}$ replaced by the partial derivatives up to order k of f.

If a probability measure π is associated with the space S and the derivative of f is required to be integrable with respect to π, then the existence and continuity of functions on sets of measure zero is not relevant. This fact is formalized by the definition of weak derivatives and Sobolev spaces. Here we will give the corresponding definitions for the multidimensional case.

Let $f \in L^1_\pi(S)$ and $\mathbf{i} = (i_1, \ldots, i_{d_S})$ be a multi-index. The $\mathbf{i}$-th order *weak derivative* $D^{(\mathbf{i})} f$ is the function in $L^1_\pi(S)$ such that

$$\int_S f \varphi^{(\mathbf{i})} \pi(\,\mathrm{d}\mathbf{x}) = (-1)^{|\mathbf{i}|} \int_S D^{(\mathbf{i})} f \varphi \pi(\,\mathrm{d}\mathbf{x}) \quad \forall \varphi \in \mathcal{C}^\infty(S) \tag{B.14}$$

The k-th *Sobolev space* associated to the probability measure π is:

$$\mathcal{H}^k_\pi(S) = \left\{ f \in L^2_\pi(S) : \sum_{|\mathbf{i}| \le k} \| D^{(\mathbf{i})} f \|_{L^2_\pi(S)} < +\infty \right\} . \tag{B.15}$$

This space is equipped with the norm $\| \cdot \|^2_{\mathcal{H}^k_\pi(S)}$ defined by

$$\| f \|^2_{\mathcal{H}^k_\pi(S)} = \sum_{|\mathbf{i}| \le k} \| D^{(\mathbf{i})} f \|^2_{L^2_\pi(S)} \tag{B.16}$$

and the semi-norm $| \cdot |_{S,\pi,k}$ defined by

$$| f |^2_{S,\pi,k} = \sum_{|\mathbf{i}| = k} \| D^{(\mathbf{i})} f \|^2_{L^2_\pi(S)}. \tag{B.17}$$

B.5 Statistical moments

Useful properties for the interpretation of random variables are their statistical moments. The *expectation* or *mean* of the random variable $X \sim \pi$ is given by

$$\mu_X = \mathbf{E}[X] = \int_{\mathbb{R}} x \, \pi(\, \mathrm{d}x) \, . \tag{B.18}$$

The (centered) *variance* of X is

$$\sigma_X^2 = \mathbf{V}[X] = \int_{\mathbb{R}} (x - \mu_X)^2 \, \pi(\, \mathrm{d}x) \, . \tag{B.19}$$

In general the n-th centered statistical moment of X is

$$\mathbf{E}[(X - \mu_X)^n] = \int_{\mathbb{R}} (x - \mu_X)^n \, \pi(\, \mathrm{d}x) \, . \tag{B.20}$$

When dealing with random vectors, the expectation is given by

$$\mu_{\mathbf{X}} = \mathbf{E}[\mathbf{X}] = (\mathbf{E}[X_1], \ldots, \mathbf{E}[X_n]) \ . \tag{B.21}$$

The second order moment of $\mathbf{X}$ is called the *covariance* and is given both by the variance of the single variables X_i, as defined in (B.19), and by combinations of two variables. The covariance is expressed in what is called the *covariance matrix*:

$$(\mathbf{C}_{\mathbf{X}})_{i,j} = \mathbf{Cov}\,[X_i, X_j] = \mathbf{E}[(X_i - \mu_{X_i})(X_j - \mu_{X_j})] \, . \tag{B.22}$$

The normalized version of the covariance is the *correlation*:

$$\mathbf{Corr}\,[X_i, X_j] = \frac{\mathbf{Cov}\,[X_i, X_j]}{\sigma_{X_i} \sigma_{X_j}} \, . \tag{B.23}$$

Two random variables X_i, X_j are called *uncorrelated* if $\mathbf{Corr}\,[X_i, X_j] = 0$. Note that two independent random variables are uncorrelated, while the converse is not true in general. In particular, if X and Y are Gaussian random variables with joint distribution π, then $\mathbf{Corr}\,[X, Y] = 0$ implies that π is a product measure – $\pi = \pi_X \times \pi_Y$ – and thus X and Y are independent.

If the random variable considered is $Y = f(X)$ where f is a real valued function and $X \sim \pi_{\mathrm{x}}$, then we will use the subscript π_{x} for all the statistical moments of f. For example, the mean of Y will be denoted by:

$$\mu_Y = \mathbf{E}\,[f]_{\pi_{\mathrm{x}}} = \int_{\mathbb{R}} f(x) \pi_{\mathrm{x}}(\, \mathrm{d}x) \tag{B.24}$$

B.6 Conditional probability and expectation

In some situations, partial information about experiments is available and probability theory makes wide use of it through conditional probabilities. For discrete probability distributions the conditional probability of event $A \in \mathscr{F}$ given $B \in \mathscr{G} \subset \mathscr{F}$ is commonly defined as

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \tag{B.25}$$

If B is an event with probability zero, we can assign any constant value to $P(A|B)$. From this it is clear that there can be many *versions* of $P(A|B)$ all agreeing but for sets of measure zero, which are irrelevant in practical situations. If B is unknown yet, we can still devise an experiment to test its occurrence: this corresponds to partitioning Ω by a countable set $\{B_i\}$ and define the experiment to be $\mathscr{G} = \sigma(\{B_i\})$. Then the conditional probability will be defined as

$$P(A|\mathscr{G}) = P(A|B_i) \qquad \text{if } \omega \in B_i, \quad i = 1, 2, \dots \tag{B.26}$$

In the general case let $\mathscr{G} \subset \mathscr{F}$ be a σ-subfield generated by the sets $\{B_i\} \in \mathscr{F}$. The experiment associated with $\mathscr{G}$ corresponds to the determination of which set of $\mathscr{G}$ contains ω. If we now fix $A \in \mathscr{F}$, we can define the finite measure $\nu : \mathscr{G} \to \overline{\mathbb{R}}_+$ by

$$\nu(G) = P(A \cap G)\,, \qquad G \in \mathscr{G} \tag{B.27}$$

Since this measure is absolutely continuous with respect to P, by the Radon-Nikodym theorem, there exists a $\mathscr{G}$-measurable and P-integrable random variable $P[A|\mathscr{G}]$ such that

$$P(A \cap G) = \nu(G) = \int_G P[A|\mathscr{G}] P(\,\mathrm{d}\omega)\,, \qquad \forall\, G \in \mathscr{G}\,. \tag{B.28}$$

$P[A|\mathscr{G}]$ is the *conditional probability* of A given $\mathscr{G}$.

The most common use of conditional probabilities is to condition one random variable against other ones. Let us consider the simplest case of $(X, Y) : \Omega \to \mathbb{R}^2$ with distribution $\pi : \sigma(X, Y) \to \overline{\mathbb{R}}_+$. Then we let $\mathscr{F} = \sigma(X) \times \mathbb{R}$ and $\mathscr{G} = \mathbb{R} \times \sigma(Y)$. If we fix $A = E \times \mathbb{R}$, for $E \in \sigma(X)$, we can define the conditional probability of A given $\mathscr{G}$ as the $\mathscr{G}$-measurable and π-integrable random variable $\pi[A|\mathscr{G}]$ defined accordingly to (B.28), which is the probability of $X \in A$ given $\mathscr{G}$. In the following we will use the shorthand $\pi[X \in A|Y]$ to denote $\pi[A|\mathscr{G}]$.

The fact of conditioning against a σ-field will not be of central importance in this work, but be aware that the simple conditioning against a random variable can lead to paradoxes, as for example the Borel's paradox [18], which is not a pathological mathematical paradox, but a rather realistic one.

Along analogous lines of thought, we can define the *conditional expectation* of X given Y as the $\mathscr{G}$-measurable and π-integrable random variable $\mathbf{E}[A|\mathscr{G}]$ identified uniquely by the Radon-Nikodym theorem as

$$\int_G \mathbf{E}[A|\mathscr{G}]\,\pi(\,\mathrm{d}\mathbf{x}) = \int_G A\pi(\,\mathrm{d}\mathbf{x})\,, \qquad G \in \mathscr{G}\,. \tag{B.29}$$

Also in this case we will use the shorthand $\mathbf{E}[X \in A|Y]$, to denote $\mathbf{E}[A|\mathscr{G}]$.

Using the definition of conditional probability, we can define the *conditional cumulative distribution function*:

$$F_{X|Y}(x) = \pi[X \leq x|Y]\,. \tag{B.30}$$

If the conditional distribution $\pi_{X|Y}$ – which exists by [18, Thm. 33.3] – admits a density with respect to the Lebesgue measure, we can define the *conditional probability density function*:

$$\rho_{X|Y}(x) = \frac{\mathrm{d}}{\mathrm{d}x} F_{X|Y}(x)\,. \tag{B.31}$$

For practical applications, if $\rho_{X,Y}(x,y)$ and $\rho_Y(y)$ are the densities of π and π_Y respectively, we have that:

$$\rho_{X|Y=y}(x) = \frac{\rho_{X,Y}(x,y)}{\rho_Y(y)}\,, \qquad \text{for } \rho_Y(y) > 0\,. \tag{B.32}$$

B.7 Stochastic processes

A *stochastic process* – also called *random process* – is a collection $\{X_i\}_{i\in I}$ of random variables on a probability space $(\Omega, \mathscr{F}, P)$. The set I can represent several things:

- $I = T = [0, \infty)$ is usually used for continuous random processes evolving in time,
- $I = T = [t_0, t_1, t_2, \ldots]$ is usually used for discrete random processes evolving in time,
- $I = [a, b]$ is frequently used for continuous random processes defined on bounded intervals of time or space, in which case the are also called *random fields*.

In the context of this work we will only use random fields denoted by the P-measurable random variable $a(\cdot, \omega) : \Omega \to L^\infty(I)$, with the set of a.e. bounded

functions as range. If we fix a particular ω, a will represent a particular *path* of the field. If instead we fix a particular $x \in I$, $a(x, \cdot) : I \to (\Omega \to \mathbb{R})$ will have the set of real valued random variables as range.

In general, random fields are not fully characterized by their *finite-dimensional distributions* $\pi_{i_1,\ldots,i_k}$, for the finite set $\{i_1, \ldots, i_k\}$ of distinct indices of I. An exception to this fact are the *Gaussian random fields*, for which

$$\mathbf{X}_{i_1,\ldots,i_k} = (a(x_{i_1}, \cdot), \ldots, a(x_{i_k}, \cdot)) \tag{B.33}$$

is a Gaussian random vector for any finite set of indices $\{i_1, \ldots, i_k\}$ in I.

Along this work we will consider random fields with finite variance, i.e. $a \in L^2(\Omega, \mathscr{F}, P; L^\infty(I))$, of which Gaussian random fields are an example, being completely determined by their first two statistical moments. In these cases it is common practice to work on processes with mean zero, defining $\tilde{a} = a - \mathbf{E}[a]$. Then the process $\tilde{a}$ is completely characterized by its *covariance function*

$$\mathbf{C}_{\tilde{a}}(x, y) = \int_\Omega \tilde{a}(x, \omega)\tilde{a}(y, \omega) P(\,\mathrm{d}\omega) \,. \tag{B.34}$$

Many analytical covariance functions exist, but the most practically used are the *squared exponential*

$$\mathbf{C}_{\tilde{a}}(x, y) = \exp\left(-\frac{|x - y|^2}{2l^2}\right) \tag{B.35}$$

and the *Ornstein-Uhlenbeck*

$$\mathbf{C}_{\tilde{a}}(x, y) = \exp\left(-\frac{|x - y|}{l}\right) \,, \tag{B.36}$$

where l is the *correlation length*, which describes how strongly the value of the field at a location $x \in I$ is correlated to locations in its neighborhood. The same definitions presented up to here can be easily rewritten for multi-dimensional arbitrary domains I.

We will say that the random field is *stationary* if the covariance $\mathbf{C}_{\tilde{a}}$ depends only on $x - y$ and not on the position of x and y in I. The random field is said to be *isotropic* if the random field depends on the Euclidean distance $|x - y|$ and not on the particular direction of $x - y$.

For further details regarding stochastic processes and their applications, the reader is referred to [18, 52].

Appendix C

Orthogonal Polynomials

Two function $f, g \in L^2_\pi(S)$ on $S \subset \mathscr{B}(\mathbb{R})$ are said to be *orthogonal* if

$$(f,g)_{L^2_\pi(S)} = \int_S f(x)g(x)\pi(\,\mathrm{d}x) = 0\,. \tag{C.1}$$

In the following we will be interested in functions which have a polynomial form $\phi_i(x) = \sum_{j=0}^{i} a_j x^j$, where i is the order of the polynomial. A set of polynomials $\{\phi_i\}_{i=0}^N \subset L^2_\pi(S)$ is said to be an *orthogonal system* if

$$(\phi_i, \phi_j)_{L^2_\pi(S)} = \gamma_i \delta_{ij}\,, \tag{C.2}$$

where δ_{ij} is the Kroneker delta function and $\gamma_i = \|\phi_i\|_{L^2_\pi(S)}$. In many cases we will rather work with *orthonormal systems* $\{\tilde{\phi}_i\}_{i=0}^N$ satisfying

$$(\tilde{\phi}_i, \tilde{\phi}_j)_{L^2_\pi(S)} = \delta_{ij}\,. \tag{C.3}$$

Orthonormal systems can easily be obtained from orthogonal systems by normalization: $\tilde{\phi}_i = \phi_i / \|\phi_i\|_{L^2_\pi(S)}$. The orthogonal/orthonormal system $\{\phi_i\}_{i=0}^\infty$ is total [55] in $L^2_\pi(S)$ and thus it forms an orthogonal/orthonormal *basis* for $L^2_\pi(S)$. On the other hand $L^2_\pi(S)$ is a separable Hilbert space, and every separable Hilbert space has a total orthogonal/orthonormal system [55].

All these definitions and results are given with respect to the measure π, which in this work will be mainly a probability measure. The most common distributions

	Distribution of Z	gPC basis polynomials	Support
Continuous	Gaussian	Hermite	$(-\infty, \infty)$
	Gamma	Laguerre	$[0, \infty)$
	Beta	Jacobi	$[a, b]$
	Uniform	Legendre	$[a, b]$
Discrete	Poisson	Charlier	$\{0, 1, 2, \dots\}$
	Binomial	Krawtchouk	$\{0, 1, 2, \dots, N\}$
	Negative binomial	Meixner	$\{0, 1, 2, \dots\}$
	Hypergeometric	Hahn	$\{0, 1, 2, \dots, N\}$

Table C.1: Correspondence between distributions and polynomial basis that ensure a strong gPC approximation.

are strictly related to some well studied polynomials. This relation is shown in table C.1. In this work our attention will be focused on continuous distribution. To this end we will present the Jacobi, the Hermite and the Laguerre polynomials in App. C.1, C.2 and C.3.

All the orthogonal polynomials can be defined by a three-terms recurrence relation [33, 201]:

$$\phi_{i+1}(x) = (A_i x + B_i)\, \phi_i(x) - C_i \phi_i(x) \,, \tag{C.4}$$

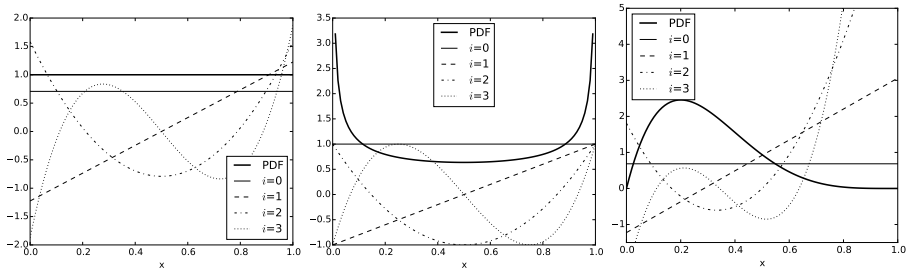
where $\phi_0 = 1$ and $\phi_1 = x$. For standard polynomials like the Jacobi, the Hermite and the Laguerre, the coefficients A_i, B_i and C_i are known analytically. The coefficients of other orthogonal polynomials with respect to non-standard measures can be approximated [33, 82]. Alternatively one can construct monomial polynomials and orthogonalize them using the Gram-Schmidt orthogonalization [73, 94] procedure.

Using the recursion coefficients A_i, B_i and C_i one can define Gauss-type quadrature rules $\mathcal{Q}_N$, based on the points and weights $\{(x_i, w_i)\}_{i=0}^N$, to approximate integrals over S:

$$\int_S g(x)\pi(\,\mathrm{d}x) \simeq \sum_{i=0}^{N} g(x_i) w_i =: \mathcal{Q}_N g \,. \tag{C.5}$$

In order to obtain the point and weights $\{(x_i, w_i)\}_{i=0}^N$ one can use the Golub-Welsch algorithm [73, 94], which requires only the knowledge of the recursion coefficients.

More details on the topic of orthogonal polynomials are available in [26, 29, 33, 202].



(a) Legendre polynomials (b) Chebyshev polynomials (c) $J^{(4,1)}$ polynomials

Figure C.1: Jacobi Polynomials with different different α and β coefficients. Left: Legendre polynomials orthogonal with respect to $\mathcal{U}([0,1])$. Center: Chebyshev polynomials orthogonal with respect to $\mathcal{B}e\left(\frac{1}{2},\frac{1}{2}\right)$. Right: $J^{(4,1)}$ polynomials orthogonal with respect to $\mathcal{B}e(2,5)$.

C.1 Jacobi Polynomials

The Jacobi polynomials are defined on the interval $S = [-1,1]$ and are orthogonal with respect to the weight function

$$w(x;\alpha,\beta) = \frac{\Gamma(\alpha+\beta+2)}{2^{\alpha+\beta+1}\Gamma(\alpha+1)\Gamma(\beta+1)}(1-x)^{\alpha}(1+x)^{\beta} \,. \tag{C.6}$$

By appropriate rescaling they can be adapted to any finite and bounded interval and in particular to the interval $[0,1]$ on which the Beta distribution is usually defined. The Beta distribution function has PDF

$$\rho_{\mathcal{B}e}(x;\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1-x)^{\beta-1} \,. \tag{C.7}$$

The relation between the weight of the Jacobi polynomials (C.6) and the PDF of the Beta distribution (C.7) is:

$$\rho_{\mathcal{B}e}(x;\alpha,\beta) = 2w(2x-1;\beta-1,\alpha-1) \,. \tag{C.8}$$

- **Recurrence relation**

$$\begin{aligned} xJ_i^{(\alpha,\beta)}(x) =& \frac{2(i+1)(i+\alpha+\beta+1)}{(2i+\alpha+\beta+1)(2i+\alpha+\beta+2)}J_{i+1}^{(\alpha,\beta)}(x) \\ &+ \frac{\beta^2-\alpha^2}{(2i+\alpha+\beta)(2i+\alpha+\beta+2)}J_i^{(\alpha,\beta)}(x) \\ &+ \frac{2(i+\alpha)(i+\beta)}{(2i+\alpha+\beta)(2i+\alpha+\beta+1)}J_{i-1}^{(\alpha,\beta)}(x) \end{aligned} \tag{C.9}$$

Figure C.1 shows the first Jacobi polynomials for different measures.

C.1.1 Legendre Polynomials

The Legendre polynomials are a special case of the Jacobi polynomials where $\alpha = \beta = 0$. When rescaled appropriately these polynomials are orthogonal with respect to the uniform $\mathcal{B}e(1,1) = \mathcal{U}([0,1])$ distribution. See Fig. C.1a.

C.1.2 Chebyshev Polynomials

The Chebyshev polynomials are a special case of the Jacobi polynomials where $\alpha = \beta = -\frac{1}{2}$. Thus on the rescaled interval $[0,1]$ they are orthogonal with respect to the $\mathcal{B}e\left(\frac{1}{2}, \frac{1}{2}\right)$. See Fig. C.1b.

C.2 Hermite Polynomials

The Hermite polynomials are defined on the real line $S := (-\infty, \infty)$. They are orthogonal with respect to measures which decay exponentially as $x \to \pm\infty$. We will see two different kinds of polynomials with this property and one associated function which is orthogonal with respect to the Lebesgue measure. Figure C.2 shows the first polynomials of these kinds.

C.2.1 Hermite Physicists' Polynomials

The Hermite Physicists Polynomials denoted by $H_i(x)$ are eigenfunctions of the Sturm-Liouville problem:

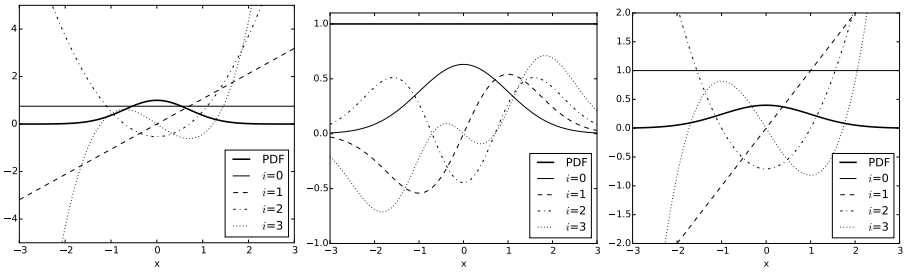
$$e^{x^2}\left(e^{-x^2}H_i'(x)\right)' + \lambda_i H_i(x) = 0, \qquad \forall x \in S := (-\infty, \infty) \tag{C.10}$$

- **Recurrence relation**

$$\begin{cases} H_0(x) = 1 \\ H_1(x) = 2x \\ H_{i+1}(x) = 2xH_i(x) - 2iH_{i-1}(x) \end{cases} \tag{C.11}$$

- **Derivatives**

$$\begin{cases} H_i^{(k)}(x) = 2iH_{i-1}^{(k-1)}(x) \\ H_i^{(0)}(x) = H_i(x) \\ H_0^{(k)}(x) = 0 \qquad \text{for } k > 0 \end{cases} \tag{C.12}$$



(a) Physicists' (b) Hermite functions (c) Probabilists'

Figure C.2: Different kinds of Hermite polynomials and functions. Left: Hermite physicists' polynomials orthogonal with respect to $w(x) = \exp(-x^2)$. Center: Hermite functions orthogonal with respect to $w(x) = 1$. Right: Hermite probabilists' polynomials orthogonal with respect to $w(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2)$.

- **Orthogonality**

$$w(x) = e^{-x^2} \tag{C.13}$$

$$\gamma_i = \sqrt{\pi} 2^i i! \tag{C.14}$$

- **Gauss Quadrature points and weights**
  The Gauss points $\{x_j\}_{j=0}^{N}$ corresponding to $H_{N+1}(x)$ can be obtained using the Golub-Welsh algorithm [73] where:

$$a_j = 0 \qquad b_j = \frac{j}{2} \tag{C.15}$$

  The Gauss weights are obtained as:

$$w_j = \frac{\lambda_N}{\lambda_{N-1}} \frac{(H_N(x), H_N(x))}{H_N(x_j) H'_{N+1}(x_j)} = \frac{\gamma_N}{(N+1) H_N^2(x_j)} \tag{C.16}$$

C.2.2 Hermite Functions

Hermite Functions are used because of their better behavior respect to Hermite Polynomials at infinity.

- **Recurrence relation**

$$\begin{cases} \tilde{H}_0(x) = e^{-x^2/2} \\ \tilde{H}_1(x) = \sqrt{2} x e^{-x^2/2} \\ \tilde{H}_{i+1}(x) = x\sqrt{\frac{2}{i+1}} \tilde{H}_i(x) - \sqrt{\frac{n}{n+1}} \tilde{H}_{i-1}(x), \qquad i \geq 1 \end{cases} \tag{C.17}$$

- **Derivatives**
  The recursion relation for the k-th derivative of the function of order n is:

$$\tilde{H}_i^{(k)}(x) = \sqrt{\frac{i}{2}}\tilde{H}_{i-1}^{(k-1)}(x) - \sqrt{\frac{i+1}{2}}\tilde{H}_{i+1}^{(k-1)}(x) \tag{C.18}$$

  Using this recursion formula we end up having an expression involving only Hermite Functions $\tilde{H}_i^{(0)}(x)$, that can be computed using the recurrence relation, and derivatives of the first Hermite Function $\tilde{H}_0^{(k)}$ that have the following form:

$$\tilde{H}_0^{(k)} = a_0 e^{-x^2/2} + a_1 x e^{-x^2/2} + a_2 x^2 e^{-x^2/2} + \ldots + a_k x^k e^{-x^2/2} \tag{C.19}$$

  The values $\{a_i\}_{i=0}^k$ can be found using the following table:

k	a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	...
0	1									...
1		-1								...
2	-1		1							...
3		3		-1						...
4	3		-6		1					...
5		-15		10		-1				...
6	-15		45		-15		1			...
7		105		-105		21		-1		...
8	105		-420		210		-28		1	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

  that can be generated iteratively using the following rules:

$$\begin{cases} A(0,0) = 1 & \\ A(i,j) = 0 & \text{if } i < j \\ A(i,j) = A(i,j) - A(i-1,j-1) & \text{if } j \neq 0 \\ A(i,j) = A(i,j) + A(i-1,j+1)(j+1) & \text{if } i > j \end{cases}$$

- **Orthogonality**

$$w(x) = 1 \tag{C.20}$$

$$\gamma_i = \sqrt{\pi} \tag{C.21}$$

- **Gauss Quadrature points and weights** The Gauss points $\{\tilde{x}_j\}_{j=0}^N$ corresponding to $\tilde{H}_{N+1}(x)$ can be obtained using the Golub-Welsh algorithm [73] where:

$$a_j = 0 \qquad b_j = \frac{j}{2} \tag{C.22}$$

These points are exactly the same of the Hermite Polynomials in (C.15). The Gauss weights are obtained as:

$$\tilde{w}_j = \frac{\gamma_N}{(N+1)\tilde{H}_N^2(x_j)} \tag{C.23}$$

C.2.3 Hermite Probabilists' Polynomials

The Hermite Probabilists' Polynomials denoted by $He_i(x)$ are eigenfunctions of the Sturm-Liouville problem:

$$\left(e^{-x^2} He_i'(x)\right)' + \lambda_i e^{-x^2} He_i(x) = 0, \qquad \forall x \in S := (-\infty, \infty) \wedge \lambda \geq 0 \tag{C.24}$$

- **Recurrence relation**

$$\begin{cases} He_0(x) = 1 \\ He_1(x) = x \\ He_{i+1}(x) = xHe_i(x) - iHe_{i-1}(x) \end{cases} \tag{C.25}$$

- **Derivatives**

$$\begin{cases} He_i^{(k)}(x) = iHe_{i-1}^{(k-1)}(x) \\ He_i^{(0)}(x) = He_i(x) \\ He_0^{(k)}(x) = 0 \qquad \text{for } k > 0 \end{cases} \tag{C.26}$$

- **Orthogonality**

$$w(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \tag{C.27}$$

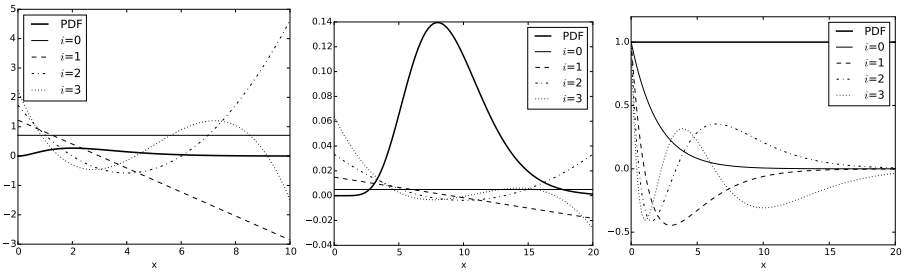
$$\gamma_i = i! \tag{C.28}$$

- **Gauss Quadrature points and weights**
  The Gauss points $\{x_j\}_{j=0}^N$ corresponding to $He_{N+1}(x)$ can be obtained using the Golub-Welsh algorithm [73] where:

$$a_j = 0 \qquad b_j = j \tag{C.29}$$

The Gauss weights are obtained as:

$$w_j = \frac{\gamma_N}{(N+1)He_N^2(x_j)} \tag{C.30}$$



(a) Laguerre poly. $\Gamma(3,1)$ (b) Laguerre poly. $\Gamma(9,1)$ (c) Laguerre functions

Figure C.3: Left: Laguerre polynomials orthogonal with respect to $\Gamma(3,1)$. Center: Laguerre polynomials orthogonal with respect to $\Gamma(9,1)$. Right: Laguerre functions relative to $\Gamma(1,1)$ and orthogonal with respect to $w(x) = 1$.

C.3 Laguerre Polynomials

The Laguerre polynomials are defined on the real half line $S := (0, \infty)$ and they are orthogonal with respect to the weight function

$$w(x;\alpha) = \frac{x^\alpha e^{-x}}{\Gamma(\alpha+1)} \qquad \text{for } \alpha > -1 \,. \tag{C.31}$$

The Gamma distribution $\Gamma(k,\theta)$ has density

$$\rho_\Gamma(x;k,\theta) = \frac{x^{k-1}\exp\left(-\frac{x}{\theta}\right)}{\Gamma(k)\theta^k} \,. \tag{C.32}$$

This means that for $\theta = 1$,

$$\rho_\Gamma(x;k,1) = w(x;k-1) \,. \tag{C.33}$$

Likewise for the Hermite polynomials, the Laguerre polynomials can be transformed into the Laguerre functions orthogonal with respect to the Lebesgue measure, by distributing the weight w. Figure C.3 shows some of these orthogonal polynomials and functions for different Gamma distributions.

- **Recurrence relation**

$$\begin{cases} \mathcal{L}_0^{(\alpha)}(x) = 1 \\ \mathcal{L}_1^{(\alpha)}(x) = \alpha + 1 - x \\ (i+1)\mathcal{L}_{i+1}^{(\alpha)}(x) = (2i+\alpha+1-x)\mathcal{L}_i^{(\alpha)}(x) - (i+\alpha)\mathcal{L}_{i-1}^{(\alpha)}(x) \qquad i \geq 2 \end{cases} \tag{C.34}$$

- **Orthogonality**

$$w(x;\alpha) = \frac{x^{\alpha} e^{-x}}{\Gamma(\alpha+1)} \,, \tag{C.31}$$

$$\gamma_i^{(\alpha)} = \frac{\Gamma(n+\alpha+1)}{\Gamma(n+1)} \tag{C.35}$$

C.3.1 Laguerre Functions

The Laguerre functions are defined as

$$\tilde{\mathcal{L}}_i^{(\alpha)}(x) := \exp\left(-\frac{x}{2}\right) \mathcal{L}_i^{(\alpha)}(x) \qquad \text{for } \alpha > -1 \,. \tag{C.36}$$

Some Laguerre functions for $\alpha = 0$ are shown in Figure C.3c.

Appendix D

Software

Along the PhD project resulting in this work, several software packages have been developed and used. All the results in this work have been obtained with software developed at the Technical University of Denmark (DTU). All the software is Open-Source and can be used for reproducing the results found in this thesis or for experimenting with new problems.

The main programming language used for the UQ part of this project is Python. Python is an interpreted high-level programming language which provides flexibility in prototyping algorithms and efficiency by the usage of an extensive C/C++/Fortran backend. The efficiency of Python is driven by the amount of work that one is able to condense on operations performed by the low-level backend. Thus the development of low-level numerical algorithms – like numerical linear algebra – is discouraged in Python, whereas the interfacing to low-level implementations is suggested and made easy by several tools, such as CPython.

Since this work focuses on non-intrusive techniques for UQ, most of the workload is always assumed to take place in the function evaluations, which should be implemented in an efficient low-level programming language. For this reason and the one listed before, Python is suitable for the job of constructing non-intrusive UQ methods, where the computational bottleneck is the function evaluation.

Three main packages and several collateral ones have been developped for UQ:

Spectral Toolbox : Construction of one and n dimensional polynomial basis

and quadrature rules. It includes also Sparse Grids and Stroud's quadrature rules. It provides also an interface to the Python package `orthpol`.

`PyPI` : https://pypi.python.org/pypi/SpectralToolbox

`Documentation` : http://pythonhosted.org/SpectralToolbox

`Code and Tracker` : https://launchpad.net/spectraltoolbox

`License` : LGPL v.3

Uncertainty Quantification Toolbox : tools for dimensionality reduction, random sampling, Polynomial Chaos (Galerkin and Collocation), HDMR, ANOVA decomposition and sensitivity analysis.

`PyPI` : https://pypi.python.org/pypi/UQToolbox

`Documentation` : http://pythonhosted.org/UQToolbox

`Code and Tracker` : https://launchpad.net/uqtoolbox

`License` : LGPL v.3

Tensor Toolbox : tensor-train decomposition, multi-linear algebra, spectral tensor-train decomposition.

`PyPI` : https://pypi.python.org/pypi/TensorToolbox

`Documentation` : http://pythonhosted.org/TensorToolbox

`Code and Tracker` : https://launchpad.net/tensortoolbox

`References` : Bigoni et al. [9]

`License` : LGPL v.3

orthpol : Python porting of the package ORTHPOL by Gautschi [82].

`PyPI` : https://pypi.python.org/pypi/orthpol

`Code and Tracker` : https://launchpad.net/pyorthpol

`License` : LGPL v.3

mpi_map : the `map` function in Python applies a input function to the elements of a list by a low-level iteration. `mpi_map` accomplishes the same but in parallel using the Message Passing Interface (MPI)

`PyPI` : `https://pypi.python.org/pypi/mpi_map`

`Code and Tracker` : https://launchpad.net/py-mpi-map

`License` : LGPL v.3

In part II of this work, some of the forward models used are the result of past projects taking place at the Technical University of Denmark:

DYnamics Train SImulation : Object oriented multi-body dynamics tool for the design and dynamical simulation of railway cars.

`Language` : C++

`Code and Tracker` : https://launchpad.net/dytsi

`References` : Bigoni [170]

`License` : LGPL v.3

OceanWave3D : The OceanWave3D is an efficient solver of the Fully Nonlinear and Dispersive Potential Flow equations. A parallel implementation of the solver for modern multi-GPU environment is also available.

`Language` : C++

`Website` : http://www2.compute.dtu.dk/ apek/OceanWave3D/

`References` : Engsig-Karup et al. [193, 194] and Glimberg et al. [203]

`License` : LGPL

D.1 Spectral Toolbox

The package `SpectralToolbox` version `0.1.9` provides the following functionalities:

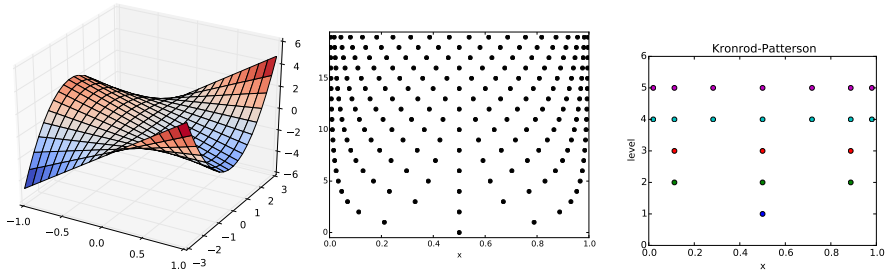
- Construction of one-dimensional polynomials (listing D.1):
  - Jacobi – see section C.1
  - Hermite physicists' – see section C.2.1
  - Hermite functions – see section C.2.2
  - Hermite probabilists' – see section C.2.3
  - Laguerre – see section C.3
  - Laguerre functions – see section C.3.1
  - Construction of orthogonal polynomials with respect to arbitrary measure (requires the package `orthpol`)
- Construction of n-dimensional polynomials and simplex bases (listing D.2).
- Construction of quadrature rules (listing D.3):
  - Gauss
  - Gauss-Lobatto
  - Gauss-Radau
- Nested quadrature rules (listing D.1):
  - Kronrod-Patterson on the real line
  - Kronrod-Patterson uniform
  - Clenshaw-Curtis
  - Fejer's
- Sparse Grids quadratures and heterogeneous quadratures (listing D.5).

Listing D.1 Jacobi polynomials shown in figure C.1a

```
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt
from SpectralToolbox import Spectral1D as S1D
N = 3
ls = ['-','--','-.',':']
x = np.linspace(0,1,100)
# Uniform distribution
alpha = 0
beta = 0
dist = stats.beta(alpha+1,beta+1)
P = S1D.Poly1D(S1D.JACOBI,[alpha,beta])
V = P.GradVandermonde1D(2*x-1,N,0,norm=True)
plt.figure(figsize=(6,5))
plt.plot(x,dist.pdf(x),'k-',linewidth=2,label='PDF')
for i in xrange(N+1):
    plt.plot(x,V[:,i],'k'+ls[i],label='$i$=%d'%i)
plt.xlabel('x')
plt.legend(loc='best')
plt.show(False)
```

Listing D.2 Simplex basis (5.10) with Jacobi times Hermite polynomials – fig. D.1a

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from SpectralToolbox import Spectral1D as S1D
from SpectralToolbox import SpectralND as SND
N = 3
# Legendre x Hermite
x = [np.linspace(-1,1,20), np.linspace(-3,3,20)]
(XX, YY) = np.meshgrid(*x)
alpha = 0
beta = 0
polys = [ S1D.Poly1D(S1D.JACOBI,[alpha,beta]), S1D.Poly1D(
    S1D.HERMITEP_PROB,None) ]
P = SND.PolyND(polys)
V = P.GradVandermondePascalSimplex( x, N, [0]*2 )
fig = plt.figure(figsize=(6,5))
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface( XX, YY, V[:,7].reshape((20,20)), rstride=1,
     cstride=1, cmap=plt.cm.coolwarm, linewidth=0,
    antialiased=False)
plt.show(False)
```



(a) Listing D.2 (b) Listing D.3 (c) Listing D.4

Figure D.1: Examples

Listing D.3 Gaussian quadrature rules for the uniform distribution – fig. D.1b

```
import numpy as np
import matplotlib.pyplot as plt
from SpectralToolbox import Spectral1D as S1D
Ns = range(20)
# Uniform distribution
alpha = 0
beta = 0
P = S1D.Poly1D(S1D.JACOBI,[alpha,beta])
plt.figure(figsize=(6,5))
for i in Ns:
    (x,w) = P.Quadrature(i,quadType=S1D.GAUSS)
    x = (x+1.)/2.
    w /= np.sum(w)
    plt.plot(x,i*np.ones(x.shape),'ko')
plt.xlabel('x')
plt.ylim([-0.5,Ns[-1]+0.5])
plt.show(False)
```

Listing D.4 Kronrod-Patterson rule – fig. D.1c

```
import numpy as np
from matplotlib import pyplot as plt
from SpectralToolbox import SparseGrids as SG
Ls = range(1,6)
plt.figure(figsize=(5,4))
for l in Ls:
    (x,w) = SG.KPU(l)
    x = np.asarray(x)
    x = np.hstack([1-x[1:][::-1],x])
    plt.plot(x,l*np.ones(len(x)),'o')
plt.ylim([0,6])
plt.xlim([0.,1.])
```

```
13 plt.show(block=False)
```

Listing D.5 Sparse Grids with Fejèr's rule – fig. 5.13b

```
1  import numpy as np
2  from matplotlib import pyplot as plt
3  from mpl_toolkits.mplot3d import Axes3D
4  from SpectralToolbox import SparseGrids as SG
5  sg = SG.SparseGrid(SG.FEJ,3,5,sym=1)
6  (sgX,sgW) = sg.sparseGrid()
7  fig = plt.figure(figsize=(5,4))
8  ax = fig.add_subplot(111, projection='3d')
9  ax.scatter(sgX[:,0], sgX[:,1], sgX[:,2], '.k')
10 plt.show(False)
```

D.2 Uncertainty Quantification Toolbox

The `uqtoolbox` version `0.1.13` provides several methods for the propagation of the uncertainty and for sensitivity analysis. We refer the reader to the examples and tests shipped along with the software. We list here the main features of the toolbox:

- Random sampling methods with MPI support through `mpi_map`:
  - Monte Carlo
  - Latin Hyper Cube
  - Quasi Monte Carlo
- Building blocks for Polynomial Chaos (gPC)
  - Galerkin
  - Collocation
- High Dimensional Model Representation
  - cut-HDMR
  - ANOVA-HDMR
- Global sensitivity analysis with the Sobol' method (listing D.6)
- Model reduction (Karhunen-Loève expansion)

Listing D.6 Computation of Sobol' indices using ANOVA-HDMR through PC based cut-HDMR, on the Sobol' g-function.

```
1  import numpy as np
2  import scipy.special as scsp
```

```
import scipy.stats as stats
from matplotlib import pyplot as plt
from SpectralToolbox import Spectral1D
from UQToolbox import CutANOVA
import UQToolbox.RandomSampling as RS

DIM = 8
pp = Spectral1D.Poly1D(Spectral1D.JACOBI,[0.,0.])
polys = [pp for i in range(DIM)]
Ns = [6 for i in range(DIM)]
cut_order = 2
X_cut = np.zeros((1,len(polys)))

tol = 2. * np.spacing(1)

cut_HDMR = CutANOVA.CutHDMR(polys,Ns,cut_order,X_cut,tol)

def fun(X,params=None):
    a = np.asarray([0., 1.,4.5, 9., 99., 99., 99., 99.]);
    if len(X.shape) == 1:
        Y = np.prod( (np.abs(4.*X - 2.) + a)/(1. + a) );
    elif len(X.shape) == 2:
        Y = np.prod( (np.abs(4.*X - 2.) + a)/(1. + a) , 1);
    return Y

def transformFunc(X):
    # from [-1,1] to [0,1]
    return (X+1.)/2.

# Evaluate f on the cutHDMR grid
print "N. eval = %d" % np.sum( [scsp.binom(DIM,i) * Ns[0]**i
    for i in range(cut_order+1)] )
cut_HDMR.evaluateFun(fun,transformFunc)
print "End evaluation"

print "Start HDMR computation"
# Compute the cutHDMR
cut_HDMR.computeCutHDMR()

# Compute the ANOVA-HDMR
cut_HDMR.computeANOVA_HDMR()
print "End HDMR computation"

# Compute an estimate for the total variance (using Monte
    Carlo)
dists = [stats.uniform(0,1) for i in xrange(DIM)]
exp_lhc = RS.Experiments(fun,None,dists,False)
```

```
48 exp_lhc.sample(2000, method='lhc')
49 exp_lhc.run()
50 MC_vals = np.asarray(exp_lhc.get_samples())
51 MC_exp = np.asarray(exp_lhc.get_results())
52 MC_mean = np.mean(MC_exp)
53 MC_var = np.var(MC_exp)
54 plt.figure()
55 plt.subplot(2,1,1)
56 plt.plot(np.array([np.mean(MC_exp[:i]) for i in range(len(
       MC_exp))]))
57 plt.subplot(2,1,2)
58 plt.plot(np.array([np.var(MC_exp[:i]) for i in range(len(
       MC_exp))]))
59
60 # Compute individual variances
61 D = []
62 for level_grids in cut_HDMR.grids:
63     D_level = []
64     for grid in level_grids:
65         D_level.append( np.dot(grid.ANOVA_HDMR_vals**2.,
              grid.WF) )
66     D.append(D_level)
67 Var_ANOVA = np.sum(D[1]) + np.sum(D[2])
68 print "TotVar/Var_Anova = %f" % (Var_ANOVA/MC_var)
69
70 # Compute Total variances per component
71 TV = np.zeros(DIM)
72 for idx in range(DIM):
73     for level, level_idxs in enumerate(cut_HDMR.idxs):
74         for j, idxs in enumerate(level_idxs):
75             if idx in idxs:
76                 TV[idx] += D[level][j]
77 TS = TV/MC_var
78 print "N      TV        TS"
79 for i,grid in enumerate(cut_HDMR.grids[1]):
80     print "%2d     %.4f    %.4f" % (i+1, TV[i], TS[i])
81
82 plt.figure()
83 plt.pie(TS/np.sum(TS),labels=["x%d"%(i+1) for i in range(DIM
       )])
84 plt.show(False)
```

D.3 Tensor Toolbox

The `TensorToolbox` is a collection of tools for the decomposition of tensors and the approximation of high-dimensional functions. Examples and unit tests for all the functionalities of the toolbox are shipped with the source code and are well documented. The `TensorToolbox` version `0.3.1` provides the following functionalities:

- TT formats
  - Tensor vectors `TTvec`
  - Tensor matrices `TTmat`
- TT construction
  - `TT-svd` [122]
  - `TT-cross` [204]
  - `TT-dmrg-cross` [134]
- Quantics TT vectors and matrices [136]
- Basic arithmetic in TT format
- Multi-linear algebra in TT format
  - Steepest descent
  - Conjugate Gradient (CG)
  - Generalized Minimal Residual method (GMRES)
- Spectral TT for functionals and fields
  - Projection
  - Interpolation

Appendix E

Included Papers

Comparison of Classical and Modern Uncertainty Qualification Methods for the Calculation of Critical Speeds in Railway Vehicle Dynamics

Daniele Bigoni, Allan P. Engsig-Karup and Hans True

DTU Informatics
The Technical University of Denmark
DK-2800 Lyngby, Denmark

Received: November 7, 2012

ABSTRACT

This paper describes the results of the application of Uncertainty Quantification methods to a railway vehicle dynamical example. Uncertainty Quantification methods take the probability distribution of the system parameters that stems from the parameter tolerances into account in the result. In this paper the methods are applied to a low-dimensional vehicle dynamical model composed by a two-axle bogie, which is connected to a car body by a lateral linear spring, a lateral damper and a torsional spring.
Their characteristics are not deterministically defined, but they are defined by probability distributions. The model - but with deterministically defined parameters - was studied in [1], and this article will focus on the calculation of the critical speed of the model, when the distribution of the parameters is taken into account.
Results of the application of the traditional Monte Carlo sampling method will be compared with the results of the application of advanced Uncertainty Quantification methods such as generalized Polynomial Chaos (gPC) [2]. We highlight the computational performance and fast convergence that result from the application of advanced Uncertainty Quantification methods. Generalized Polynomial Chaos will be presented in both the Galerkin and Collocation form with emphasis on the pros and cons of each of those approaches.

Keywords: railway vehicle dynamics, nonlinear dynamics, uncertainty quantification, generalized polynomial chaos, high-order cubature rules.

1. INTRODUCTION

In the engineering field, deterministic models have been extensively exploited to describe dynamical systems and their behaviors. These have proven to be useful in the design phase of the engineering production, but they always fell short in providing indications of the reliability of certain designs over others. The results obtained by one deterministic experiment describe, in practice, a very rare case that likely will never happen. However, we are confident that this experiment will explain most of the experiments in the vicinity of it, i.e. for small variation of parameters. This assumption is wrong, in particular for realistic nonlinear dynamical systems, where small perturbations can cause dramatic changes in the dynamics. It is thus critical to find a measure for the level of our knowledge of a dynamical system, in order to be able to make reasonable risk analysis and design optimization.

Risk analysis in the railway industry is critical for as well the increase of the safety as for targeting investments. Railway vehicle dynamics are hard to study even in the deterministic case, where strong nonlinearities appear in the system. A lot of phenomena

develop within such dynamical systems and the interest of the study could be focused on different parameters, such as ride comfort or wear of the components. This work will instead focus on ride safety when high-speeds are reached and the hunting motion develops. The hunting motion is a well known phenomenon characterized by periodic as well as chaotic lateral oscillations, due to the wheel-rail contact forces, that can appear at different speeds depending on the vehicle design. This motion can be explained and studied with notions from nonlinear dynamics [3], as well as suitable numerical methods for non-smooth dynamical systems [4]. It is well known that the behavior of the hunting motion is parameter dependent, thus good vehicle designs can increase the critical speed where the hunting motion starts. This also means that suspension components need to be carefully manufactured in order to really match the constructor's expectations. However, no manufactured component will ever match the simulated ones. Thus epistemic uncertainties, for which we have no evidence, and aleatoric uncertainties, for which we have a statistical description, appear in the system as a level of knowledge of the real parameters [5].

Uncertainty quantification (UQ) tries to address the question: "assuming my partial knowledge of the design parameters, how reliable are my results?". The UQ field can then be split in the study of rare events (e.g. breaking probability), that develop at the tails of probability distributions, and the study of parameter sensitivity, that focus on events with high probability. This work will focus on the sensitivity of the critical speed of a railway vehicle model to the suspension parameters.

2. THE VEHICLE MODEL

This work will investigate the dynamics of the well known Cooperrider model [1] shown in Fig. 1. The model is composed by two conical wheel sets rigidly connected to a bogie frame, that is in turn connected to a fixed car body by linear suspensions: a couple of lateral springs and dampers and one torsional spring.

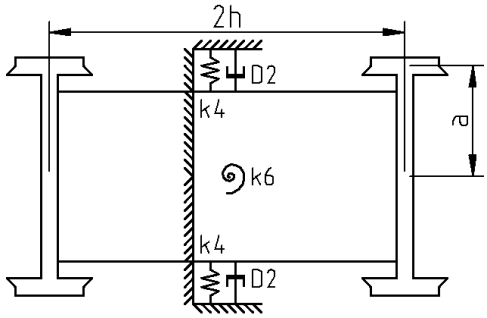


Fig. 1: Top view of the Cooperrider bogie model.

We use the governing equations of this dynamical system as in [1]:

$$
\begin{aligned}
m\ddot{q}_1 &= -2D_2\dot{q}_1 - 2k_4q_1 - 2\left[F_x\left(\xi_{x_1},\xi_{y_1}\right) + F_x\left(\xi_{x_2},\xi_{y_2}\right)\right] - F_T(q_1 + haq_2) \\
&\quad - F_T(q_1 - haq_2), \\
I\ddot{q}_2 &= -k_6q_2 - 2ha\left[F_x\left(\xi_{x_1},\xi_{y_1}\right) - F_x\left(\xi_{x_2},\xi_{y_2}\right)\right] - 2a\left[F_y\left(\xi_{x_1},\xi_{y_1}\right) + F_y\left(\xi_{x_2},\xi_{y_2}\right)\right] \\
&\quad - ha[F_T(q_1 + haq_2) - F_T(q_1 - haq_2)],
\end{aligned}
\tag{1}
$$

where D_2, k_4 and k_6 are the damping coefficient and the stiffness coefficients respectively, F_x and F_y are the lateral and longitudinal creep forces and F_T is the flange force.

The ideally stiff bogie runs on a perfect straight track where the constant wheel-rail friction enters the system through the lateral and longitudinal creep-forces:

$$
F_x\left(\xi_x,\xi_y\right) = \frac{\xi_x}{\phi}\frac{F_R(\xi_x,\xi_y)}{\xi_R(\xi_x,\xi_y)} \ , \quad F_y\left(\xi_x,\xi_y\right) = \frac{\xi_y}{\psi}\frac{F_R(\xi_x,\xi_y)}{\xi_R(\xi_x,\xi_y)} \ ,
$$

$$
\xi_R\left(\xi_x,\xi_y\right) = \sqrt{\frac{\xi_x^2}{\phi^2} + \frac{\xi_y^2}{\psi^2}} \ ,
$$

$$
\frac{F_R\left(\xi_x,\xi_y\right)}{\mu N} = \begin{cases} u(\xi_R) - \dfrac{1}{3}u^2(\xi_R) + \dfrac{1}{27}u^3(\xi_R) & \text{for } u(\xi_R) < 3 \\ 1 & \text{for } u(\xi_R) \geq 3 \end{cases} ,
$$

$$
u(\xi_R) = \frac{G\pi ab}{\mu N}\xi_R \ ,
$$

where the creepages are given by:

$$
\xi_{x_1} = \frac{\dot{q}_1}{v} + ha\frac{\dot{q}_2}{v} - q_2 \ , \qquad \xi_{y_1} = a\frac{\dot{q}_2}{v} + \frac{\lambda}{r_0}(q_1 + haq_2) \ ,
$$

$$
\xi_{x_2} = \frac{\dot{q}_1}{v} - ha\frac{\dot{q}_2}{v} - q_2 \ , \qquad \xi_{y_2} = a\frac{\dot{q}_2}{v} + \frac{\lambda}{r_0}(q_1 - haq_2) \ .
$$

The flange forces are approximated by a very stiff non-linear spring with a dead band:

$$
F_T(x) = \begin{cases} \exp\left(-\alpha/\left(x - x_f\right)\right) - \beta x - \kappa \, , & 0 \leq x < b \\ k_0 \cdot (x - \delta) \, , & b \leq x \\ -F_T(-x) \, , & x < 0 \end{cases} ,
$$

The parameters used for the analysis are listed in the following:

$m = 4963\ kg$	$h = 1.5\ m$	$a = 0.7163\ m$
$I = 8135\ kg \cdot m^2$	$D_2 = 29200\ N \cdot s/m$	$k_0 = 14.60 \cdot 10^6\ N/m$
$k_4 = 0.1823 \cdot 10^6\ N/m$	$k_6 = 2.710 \cdot 10^6\ N/m$	$\lambda = 0.05$
$r_0 = 0.4572\ m$	$b = 0.910685 \cdot 10^{-2}\ m$	$\phi = 0.60252$
$\psi = 0.54219$	$G\pi ab = 6.563 \cdot 10^6\ N$	$\mu N = 10^4\ N$
$\delta = 0.0091\ m$	$\alpha = 0{,}1474128791 \cdot 10^{-3}$	$\beta = 1{,}016261260$
$\kappa = 1{,}793756792$	$x_f = 0.9138788366 \cdot 10^{-2}$	

2.1 Non linear dynamics of the deterministic model

The dynamics of the deterministic model at high speed have been illustrated in [1]. The existence of a subcritical Hopf-bifurcation has been detected at $v_L = 66.6107\ m/s$. Fig. 2 shows the entire bifurcation diagram of the deterministic system. The linear critical speed is obtained by observation of the stability of the trivial solution using the eigenvalues of the Jacobian of the system. The nonlinear critical speed, characteristic in subcritical Hopf-bifurcations, is found at $\mathrm{v_{NL}} = 62.0206\ \mathrm{m/s}$ using a ramping method, where the speed is quasi-statically decreased, according to

$$\dot{v} = \begin{cases} 0\,, & \text{if } t < t_{st} \vee \|\vec{q}\|_2 < \epsilon_{min} \\ -\Delta\,, & \text{otherwise} \end{cases} . \tag{2}$$

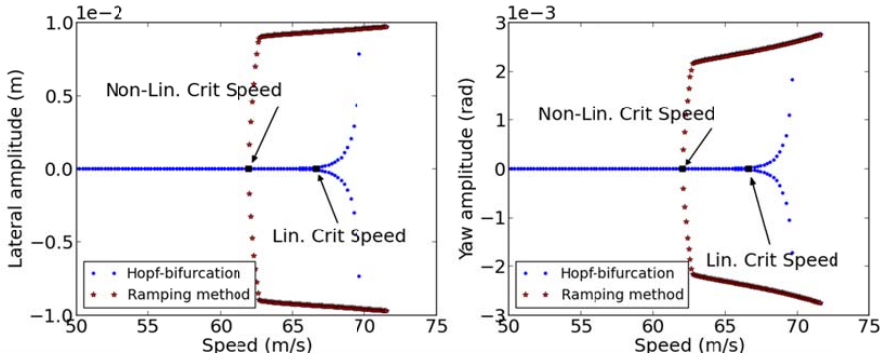


Fig. 2: Non-linear dynamics of the deterministic system. The subcritical Hopf-bifurcation is highlighted and the critical speed is determined exactly at $v_L = 66.6107\ m/s$. The ramping method is then used in order to detect the non-linear critical speed at $v_{NL} = 62.0206\ m/s$.

2.2 The stochastic model

Let's now consider that the suspensions are provided by the manufacturer with a certain level of working accuracy. In this initial study we will use Gaussian distributions to describe such uncertainties:

$$\begin{aligned} k_6 &\sim \mathcal{N}\left(\mu_{k_6}, \sigma_{k_6}^2\right) = \mathcal{N}\left(2.71 \cdot 10^6\ \frac{N}{m}, 1.84 \cdot 10^{10}\ \left(\frac{N}{m}\right)^2\right)\ , && \text{(std. } \sim 5\%) \\ k_4 &\sim \mathcal{N}\left(\mu_{k_4}, \sigma_{k_4}^2\right) = \mathcal{N}\left(9.12 \cdot 10^4\ \frac{N}{m}, 4.15 \cdot 10^7\ \left(\frac{N}{m}\right)^2\right)\ , && \text{(std. } \sim 7\%) \\ D_2 &\sim \mathcal{N}\left(\mu_{D_2}, \sigma_{D_2}^2\right) = \mathcal{N}\left(1.46 \cdot 10^4\ N \cdot \frac{s}{m}, 1.07 \cdot 10^6\ \left(N \cdot \frac{s}{m}\right)^2\right)\ . && \text{(std. } \sim 7\%) \end{aligned} \tag{3}$$

where the symmetry of the model was considered in parameters k_4 and D_2.

Now the deterministic model is turned into a stochastic model, where the single solution represents a particular realization and probabilistic moments can be used to describe the statistics of the stochastic solution.

3. UNCERTAINTY QUANTIFICATION

The stochastic solution of the system is now represented by $\boldsymbol{q}(t,\boldsymbol{Z})$, where $\boldsymbol{Z}$ is a vector of random variables distributed according to (3). We can think about it as a function that spans over a three dimensional random space. In this work we will restrict our interest in the first few moments of this solution, namely the mean $\boldsymbol{E}[\boldsymbol{q}(t,\boldsymbol{Z})]$ and variance $\boldsymbol{V}[\boldsymbol{q}(t,\boldsymbol{Z})]$, but the following is valid for higher moments too. Mean and variance are defined as

$$\begin{aligned}\boldsymbol{\mu_q}(t) &= \boldsymbol{E}[\boldsymbol{q}(t,\boldsymbol{Z})]_{\rho_{\boldsymbol{Z}}} = \iiint \boldsymbol{q}(t,\boldsymbol{z})\rho_{\boldsymbol{Z}}(\boldsymbol{z})d\boldsymbol{z} \ , \\ \boldsymbol{\sigma_q^2}(t) &= \boldsymbol{V}[\boldsymbol{q}(t,\boldsymbol{Z})]_{\rho_{\boldsymbol{Z}}} = \iiint \left(\boldsymbol{q}(t,\boldsymbol{z}) - \mu_{\boldsymbol{q}}(t)\right)^2 \rho_{\boldsymbol{Z}}(\boldsymbol{z})d\boldsymbol{z} \ ,\end{aligned} \tag{4}$$

where $\rho_{\boldsymbol{Z}}(\boldsymbol{z})$ is the probability density function of the random vector $\boldsymbol{Z}$ and the integrals are computed over its domain.

A straightforward way of computing the moments of the solution is to approximate the integrals as:

$$\begin{aligned}\boldsymbol{\mu_q}(t) &\approx \bar{\boldsymbol{\mu}}_{\boldsymbol{q}}(t) = \frac{1}{M}\sum_{j=1}^{M} \boldsymbol{q}\left(t,\boldsymbol{Z}^{(j)}\right) \ , \\ \boldsymbol{\sigma_q^2}(t) &\approx \bar{\boldsymbol{\sigma}}_{\boldsymbol{q}}^2(t) = \frac{1}{M-1}\sum_{j=1}^{M} \left(\boldsymbol{q}\left(t,\boldsymbol{Z}^{(j)}\right) - \bar{\mu}_{\boldsymbol{Z}}(t)\right)^2 \ ,\end{aligned} \tag{5}$$

where $\left\{\boldsymbol{Z}^{(j)}\right\}_{J=1}^{M}$ are realizations sampled randomly from the probability distribution of $\boldsymbol{Z}$. This is the Monte-Carlo (MC) method and it has a probabilistic error of $\mathcal{O}\left(1/\sqrt{M}\right)$.

Even if MC methods are really robust and versatile, such a slow convergence rate is problematic when the solution of a single realization of the system is computationally expensive. Alternative sampling methods are the Quasi Monte-Carlo methods (QMC). These can provide convergence rates of $\mathcal{O}((\log M)^d/M)$, where d is the dimension of the random space. They use low discrepancy sequences in order to uniformly cover the sampling domain. Without presumption of completeness, in this work we will consider only the Sobol sequence as a measure of comparison with respect to other advanced UQ methods. QMC methods are known to work better than MC methods when the integrand is sufficiently smooth, whereas they can completely fail on an integrand of unbounded variation [6]. Furthermore, randomized versions of the QMC method are available in order to improve the variance estimation of the method.

3.1 Generalized Polynomial Chaos (gPC)

Polynomial Chaos was first used by Wiener studying the decomposition of Gaussian processes [7]. It has been recently extended by Xiu for generalized distribution functions [2]. The idea is to expand the input parameters with respect to a set of N orthogonal polynomials that span P_N^d and seek a solution such that its residue is orthogonal to P_N^d. Depending on the knowledge of the analytical form of $\rho_{\boldsymbol{Z}}(\boldsymbol{z})$ a strong convergence (e.g. in the L^2-norm) or a weak convergence (in probability) can be achieved. Furthermore, given

the projection operator $\pi_N: L^2_\omega(\mathcal{R}) \to P^d_N$, with measure ω, the following result holds for unbounded domains [8]:

$$\|\boldsymbol{q} - \pi_N(\boldsymbol{q})\|_{L^2_\omega} \leq C N^{-\frac{p}{2}} \|\boldsymbol{q}\|_{H^p_\omega} \tag{6}$$

where $\left(H^p_\omega, \|\cdot\|_{H^p_\omega}\right)$ is the Sobolev space and p is its order.

For Gaussian random variables, strong convergence is guaranteed by the Hermite probabilists' polynomials:

$$\begin{aligned} &\mathcal{H}_{n+1}(x) = x\mathcal{H}_n(x) - n\mathcal{H}_{n-1}(x), \quad n > 0 \ , \\ &\int_{-\infty}^{\infty} \mathcal{H}_m(x)\mathcal{H}_n(x) \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx = \gamma_n \delta_{nm} = n!\, \delta_{nm} \ . \end{aligned} \tag{7}$$

Thus, let's consider the set of basis $\{\mathcal{H}_k(\boldsymbol{Z})\}_{|k|\leq N}$, where k is a multi-index, that span the 3-dimensional random space up to the polynomial order N and let $\boldsymbol{\alpha}(\boldsymbol{Z}) = \boldsymbol{\mu} + \boldsymbol{\sigma}\boldsymbol{Z}$ be the parameterization of the random space where $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the vectors of means and standard deviations of the input parameters. We can now rewrite the random input and the solution as:

$$\begin{aligned} \boldsymbol{\alpha}_N(\boldsymbol{Z}) &= \sum_{0\leq|k|\leq N} \widehat{\boldsymbol{\alpha}}_k \, \mathcal{H}_k(\boldsymbol{Z}) \, , \quad \widehat{\boldsymbol{\alpha}}_k = \frac{1}{\gamma_k} \iiint \boldsymbol{\alpha}(\boldsymbol{z}) \, \mathcal{H}_k(\boldsymbol{z}) \rho_{\boldsymbol{Z}}(\boldsymbol{z}) d\boldsymbol{z} \ , \\ \boldsymbol{q}_N(t, \boldsymbol{Z}) &= \sum_{0\leq|k|\leq N} \widehat{\boldsymbol{q}}_k(t) \, \mathcal{H}_k(\boldsymbol{Z}) \, , \quad \widehat{\boldsymbol{q}}_k(t) = \frac{1}{\gamma_k} \iiint \boldsymbol{q}(t, \boldsymbol{z}) \, \mathcal{H}_k(\boldsymbol{z}) \rho_{\boldsymbol{Z}}(\boldsymbol{z}) d\boldsymbol{z} \ . \end{aligned} \tag{8}$$

We then seek $\boldsymbol{q}_N(t, \boldsymbol{Z})$ that for all $|k| \leq N$ satisfies the Galerkin formulation

$$\begin{cases} \boldsymbol{E}[\partial_t \boldsymbol{q}_N(t, \boldsymbol{Z}) \, \mathcal{H}_k(\boldsymbol{Z})]_{\rho_{\boldsymbol{Z}}} = \boldsymbol{E}\left[\mathcal{L}\left(\boldsymbol{q}_N(t, \boldsymbol{Z})\right) \mathcal{H}_k(\boldsymbol{Z})\right]_{\rho_{\boldsymbol{Z}}}, & (0, T] \\ \widehat{\boldsymbol{q}}_k(0) = \widehat{\boldsymbol{q}}_{0,k}, & t = 0 \end{cases} \tag{9}$$

where the expectation operator is the projection with measure $\rho_{\boldsymbol{Z}}(\boldsymbol{z})$ and $\mathcal{L}$ is the operator defined by the right hand side of the deterministic equation. This gives a system of $K = \sum_{i=0}^{N} \binom{i + (d-1)}{d-1}$ coupled equations that can be treated with standard ODE solvers. The moments of the solution can then be recovered by:

$$\begin{aligned} \boldsymbol{\mu}_{\boldsymbol{q}}(t) &\approx \boldsymbol{E}[\boldsymbol{q}_N(t, \boldsymbol{Z})]_{\rho_{\boldsymbol{Z}}} = \widehat{\boldsymbol{q}}_0(t) \ , \\ \boldsymbol{\sigma}^2_{\boldsymbol{q}}(t) &\approx \boldsymbol{V}[\boldsymbol{q}_N(t, \boldsymbol{Z})]_{\rho_{\boldsymbol{Z}}} = \sum_{1\leq|k|\leq N} \boldsymbol{\gamma}_k \, \widehat{\boldsymbol{q}}^2_k(t) \ . \end{aligned} \tag{10}$$

3.2 Stochastic Collocation Method (SCM)

Collocation methods require the residual of the governing equations to be zero at the collocation points $\{\boldsymbol{Z}^{(j)}\}_{j=1}^{Q}$, i.e.

$$\begin{cases} \partial_t \boldsymbol{q}\left(t, \boldsymbol{Z}^{(j)}\right) = \mathcal{L}\left(\boldsymbol{q}\left(t, \boldsymbol{Z}^{(j)}\right)\right), & (0, T] \\ \boldsymbol{q}(0) = \boldsymbol{q}_0, & t = 0 \end{cases} \tag{11}$$

Then we can find $\boldsymbol{w}(t, \boldsymbol{Z})$ in the polynomial space $\Pi(\boldsymbol{Z})$ that approximates $\boldsymbol{q}(t, \boldsymbol{Z})$. We

can for instance use projection rules over a set of Hermite polynomials, so that:

$$\boldsymbol{w}_N(t,\boldsymbol{Z}) = \sum_{|k|\leq N} \widehat{\boldsymbol{w}}_k(t)\,\mathcal{H}_k(\boldsymbol{Z}) \ ,$$
$$\widehat{\boldsymbol{q}}_k = \frac{1}{\gamma_k}\iiint \boldsymbol{q}(t,\boldsymbol{z})\,\mathcal{H}_k(\boldsymbol{z})\rho_{\boldsymbol{Z}}(\boldsymbol{z})d\boldsymbol{z} \approx \widehat{\boldsymbol{w}}_k = \frac{1}{\gamma_k}\sum_{j=1}^{Q} \boldsymbol{q}\left(t,\boldsymbol{z}^{(j)}\right)\mathcal{H}_k\left(\boldsymbol{z}^{(j)}\right)\alpha^{(j)} \ , \tag{12}$$

where we used a cubature rule with points and weights $\left\{\boldsymbol{z}^{(j)},\alpha^{(j)}\right\}_{j=1}^{Q}$. Cubature rules with different accuracy levels and sparsity exist. In this work we will use simple tensor product structured Gauss cubature rules, that are the most accurate but scale with $\mathcal{O}(m^d)$, where m is the number of points in one dimension and d is the dimension of the random space. The fast growth of the number of collocation points with the dimensionality goes under the name of "curse of dimensionality" and can be addressed using more advanced cubature rules such as Smolyak sparse grids [9].

4. UNCERTAINTY QUANTIFICATION ON RAILWAY VEHICLE DYNAMICS

Uncertainty quantification is recently gaining much attention from many engineering fields and in vehicle dynamics we can already find some contributions on the topic. In [10] a railway vehicle dynamic problem with uncertainty on the suspension parameters was investigated using MC method coupled with techniques from Design of Experiments. In [11] gPC was first applied to a linearized model of a simple vehicle on uneven terrain.

Here gPC and SCM will be applied to the simple Cooperrider bogie frame in order to study its behavior with uncertainties and the results will be compared to the one obtained by the MC method. Fig. 3 shows the application of gPC of order 5 on the model running at constant speed. The method solves a system of 140 coupled equations and is able to approximate the first two moments of the solution.

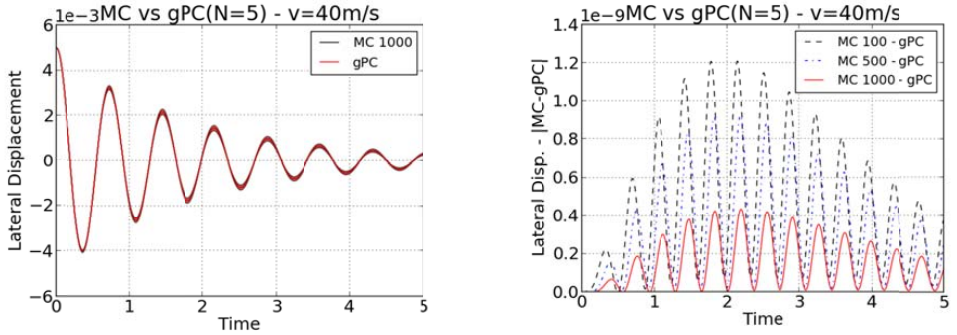


Fig. 3: (Right) Mean and Variance of the stochastic solution for the lateral displacement of the model running at constant speed. (Left) Convergence of Monte Carlo variance to the gPC solution. 100, 500 and 1K realizations has been used for Monte Carlo method.

Comparable results can be obtained using SCM where 216 collocation points are used. Both the methods perform well as long as the solution is sufficiently smooth in the

random space. This is clearly not the case when bifurcations occur and different realizations of random parameters determine different attractors for the solutions. In this case the spectral convergence expected from gPC will drop to linear convergence.

The focus of this work is on the determination of the nonlinear critical speed with uncertainties, so the investigation of the stochastic dynamics w.r.t. time will be disregarded here. Fig. 4 shows the SCM method applied to the model with 1D uncertainty on parameter k_4, for the determination of the first two moments of the nonlinear critical speed. The estimation done by the SCM is already satisfactory at low order and little is gained by increasing it. This means that the few first terms of the expansion (12) are sufficient in approximating the nonlinear critical speed distribution.

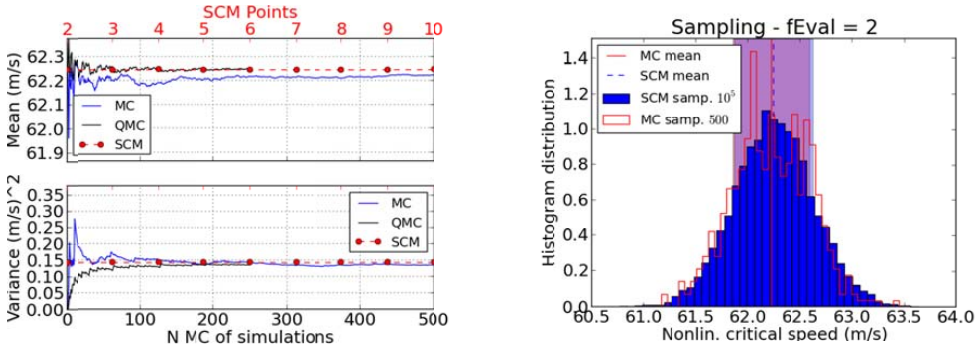


Fig. 4: SCM on the model with 1D uncertainty on parameter k_4. Left, estimation of mean and variance of the nonlinear critical speed. Right, histograms of NL critical speeds obtained using 500 MC simulations of model (1)-(2) *and* 10^5 *realizations using the approximated stochastic solution* (12) *with only 2 function evaluations. The standard deviation is shown as a shaded confidence interval, blue for SCM and red for MC.*

Fig. 5 shows the SCM method applied to the same problem with 1D uncertainty on the torsional spring stiffness k_6. Again the first few terms in expansion (12) are sufficient in order to give a good approximation of the nonlinear critical speed distribution. We can also notice that the torsional spring stiffness k_6 has an higher influence on the critical speed than k_4.

Fig. 6 shows the SCM method on the problem with uncertainty on parameters k_6, k_4 and D_2. Again we see that the a low-order SCM approximation is sufficient to get the most accurate solution.

Table 1 shows the final results with maximum accuracy, obtained using the three methods. We can observe that the variances in the multiple-dimensional cases are almost equal to the sum of the single-dimensional cases. This means that there is no nonlinear effect appearing due to the consideration of multiple uncertainties in this case.

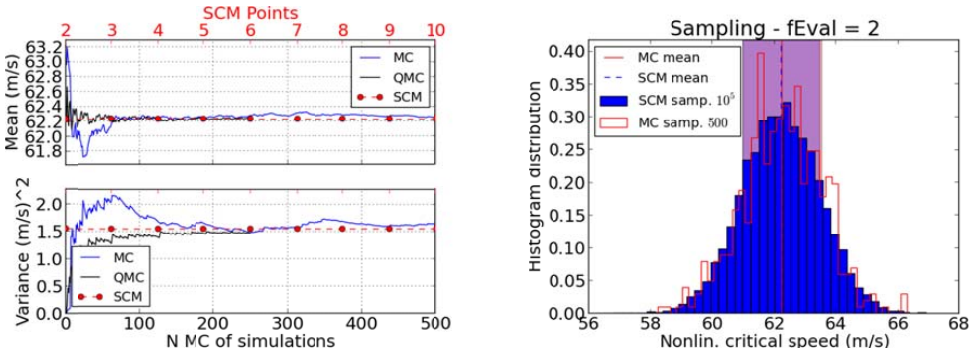


Fig. 5: SCM on 1D uncertainty. Left, estimation of mean and variance of the non-linear critical speed. Right, histograms of NL critical speeds obtained using 500 MC simulations of model (1)-(2) *and 500 realizations using the approximated stochastic solution* (12) *with only 2 function evaluations. The standard deviation is shown as a shaded confidence interval, blue for SCM and red for MC.*

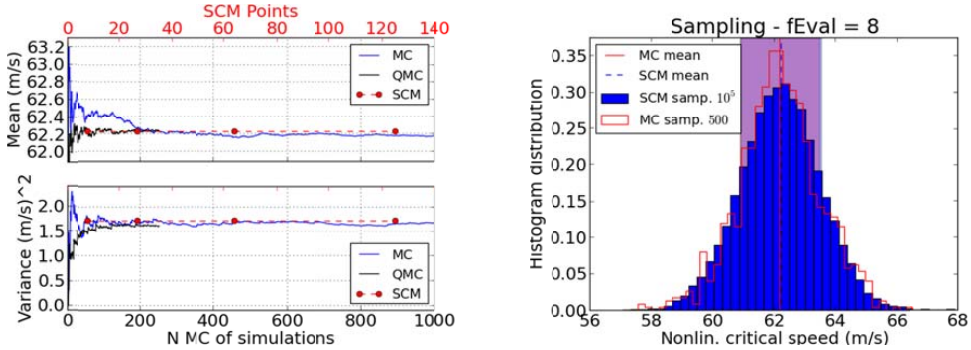


Fig. 6: SCM on 3D uncertainty. Left, estimation of the mean and variance of the non-linear critical speed. Right, histograms of nonlinear critical speeds.

	MC (500 eval.)		**QMC** (250 eval)		**SCM** (max. order)	
	$\boldsymbol{\mu}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\mu}$	$\boldsymbol{\sigma^2}$	$\boldsymbol{\mu}$	$\boldsymbol{\sigma^2}$
k_6	62,259304	1,6427635	62,244701	1,4731305	62,229081	1,5544725
k_4	62,225047	0,1361424	62,251760	0,1359384	62,247742	0,1431684
D_2	62,234186	0,0248543	62,251042	0,0238190	62,248916	0,0250455
k_6, k_4	62,222645	1,5339337	62,223559	1,6168049	62,281098	1,6861942
k_6, D_2	62,176463	1,7153238	62,244451	1,4967552	62,281913	1,5677285
k_4, D_2	62,247277	0,1739290	62,250638	0,1597684	62,301024	0,1690544
k_6, k_4, D_2[1]	62,183424	1,6806237	62,233991	1,6287313	62,229247	1,7236020

Table 1: Estimated mean and variance of the nonlinear critical speed using MC, QMC and SCM.

[1] For the full 3D uncertainty problem, the number of evaluation used for MC has been increased to 10^3.

5. CONCLUSIONS

Two approaches to the stochastic treatment of a railway dynamical system have been presented. MC doesn't make any assumption on the regularity of the stochastic solution, thus it is outperformed by QMC, gPC and SCM, when a certain level of smoothness is present. In particular gPC and SCM can be 100 times faster than MC for low-dimensional problems. For high-dimensional problems gPC/SCM methods suffer from the "curse of dimensionality". Techniques, such as sparse grids [9], are available to reduce this effect, but these all rely on the smoothness of the solution and in most cases only work for standard distributions.

We have shown how modern techniques for UQ can improve efficiency in the computation of statistics for models with a limited number of uncertainties. This represents a useful tool for engineers during the design phase, where potential risks due to uncertainties can be readily detected.

6. REFERENCES

[1] H. True and C. Kaas-Petersen, "A Bifurcation Analysis of Nonlinear Oscillations in Railway Vehicles," *Vehicle System Dynamics* , 1983.

[2] D. Xiu, Numerical Methods for Stochastic Computations: A Spectral Method Approach, Princeton: Princeton University Press, 2010.

[3] H. True, "On the Theory of Nonlinear Dynamics and its Applications in Vehicle Systems Dynamics," *Vehicle System Dynamics* , vol. 31, pp. 393-421, 1999.

[4] H. True, A. P. Engsig-Karup and D. Bigoni, "On the Numerical and Computational Aspects of Non-Smoothnesses that occur in Railway Vehicle Dynamics," *Mathematics and Computers in Simulation*, 2012.

[5] S. F. Wojtkiewicz, M. S. Eldred, R. V. Field, A. Urbina and J. R. Red-Horse, "Uncertainty Quantification In Large Computational Engineering Models," *American Institute of Aeronautics and Astronautics* , vol. 14, 2001.

[6] W. J. Morokoff and R. E. Caflisch, "Quasi-Monte Carlo Integration," *Journal of Computational Physics* , vol. 122, no. 2, pp. 218-230 , 1995.

[7] N. Wiener, "The homogeneous chaos," *American Journal of Mathematics*, vol. 60, no. 4, pp. 897-936, 1938.

[8] J. Shen and L. L. Wang, "Some recent advances on spectral methods for unbounded domains," *Communications in Computational Physics* , vol. 5, no. 2-4, pp. 195-241, 2009.

[9] K. Petras, "Smolyak cubature of given polynomial degree with few nodes for increasing dimension," *Numerische Mathematik* , vol. 93, no. 4, pp. 729-753 , 2003.

[10] L. Mazzola and S. Bruni, "Effect of Suspension Parameter Uncertainty on the Dynamic Behaviour of Railway Vehicles," *Applied Mechanics and Materials* , vol. 104, pp. 177-185 , 2011.

[11] G. Kewlani, J. Crawford and K. Iagnemma, "A polynomial chaos approach to the analysis of vehicle dynamics under uncertainty," *Vehicle System Dynamics*, vol. 50, no. 5, pp. 749-774 , 2012.

Anwendung der „Uncertainty Quantification“ bei eisenbahndynamischen Problemen

Application of the „Uncertainty Quantification“ in railway dynamical problems

M.Sc. Daniele Bigoni, Professor Allan P. Engsig-Karup und Professor em. Hans True

Zusammenfassung

Die Anwendung von 'Uncertainty Quantification Methoden' in der Eisenbahnfahrzeugdynamik wird präsentiert. Die Systemparameter sind durch Verteilungsfunktionen gegeben. Die Ergebnisse der Anwendung von Monte-Carlo und 'generalized Polynomial Chaos' Methoden auf einem einfachen Drehgestell Modell wird diskutiert.

Abstract

The paper describes the results of the application of Uncertainty Quantification methods in railway vehicle dynamics. The system parameters are given by probability distributions. The results of the application of the Monte-Carlo and generalized Polynomial Chaos methods to a simple bogie model will be discussed.

1. Einführung.

Bei den theoretischen Untersuchungen in der Eisenbahnfahrzeugdynamik werden für die Anwendungen dynamische Modelle mit größter Sorgfalt mathematisch formuliert. Für die Parameter in den Problemen wie Adhäsionsbeiwert und die Charakteristiken der Aufhängung werden Festwerte oder wohldefinierte Funktionen gewählt um zu einem deterministischen System zu gelangen. Die Analyse der Probleme der Eisenbahnfahrzeugdynamik ist sowieso wegen der Anzahl der Körper und der damit verbundenen vielen Freiheitsgrade, und der Nichtlinearität und Nicht-Glattheit vieler Funktionen schwierig. Im Fall der Untersuchung eines schon existierenden Fahrzeugs müssen alle Parameterwerte und funktionale Zusammenhänge in dem dynamischen Problem von vornherein gemessen werden, was aber beim Entwurf eines Fahrzeugs nicht möglich ist. Beim Entwurf werden deswegen die nominellen Werte und Funktionen in das dynamische Problem substituiert, dessen Lösung dann als Grundlage für die Bewertung der dynamischen Eigenschaften des Fahrzeugs dient. Viele Werte wie auch Funktionen werden paarweise gleich gesetzt, wobei sehr oft mindestens eine Symmetrie, nämlich die um die Längsachse des Eisenbahnfahrzeugs, in das dynamische Problem eingeführt wird. Die Symmetrie spielt eine Rolle für die Existenz der Lösungen des dynamischen Problems. Zum Beispiel haben symmetrische nichtlineare dynamische Probleme im Allgemeinen keine chaotischen Lösungen, die dann erst nach einer symmetriebrechenden Verzweigung bei höheren Geschwindigkeiten existieren können.
Alle Elemente in der Fahrzeugaufhängung werden aber mit Fertigungstoleranzen hergestellt, und erstens existiert deswegen kein symmetrisch gebautes Eisenbahnfahrzeug, und zweitens sind die Lösungen des deterministischen dynamischen Problems mit den wohldefinierten Parameterwerten im Vergleich zur Dynamik des wirklichen Fahrzeugs eine Annäherung, deren Güte nicht abgeschätzt ist. Der Ingenieur hat dabei ein ungutes Gefühl. Wie zuverlässig sind die Rechenergebnisse – z.B. eine Berechnung der kritischen Geschwindigkeit eines Eisenbahnfahrzeugs? Er kann sich nur auf Erfahrungen mit früheren Berechnungen stützen, aber halten die Ergebnisse wirklich auch für ein neues Fahrzeug mit seinen noch nicht in der Konstruktion geprüften Elementen?
Eine Abschätzung kann mit Hilfe der Monte-Carlo Methode wie in der Arbeit von Mazzola und Bruni [1] durchgeführt werden. Ihre Berechnungen brauchen aber viel Rechenzeit, und wir schlagen deswegen eine andere Methode vor, die weniger zeitaufwendig ist und mindestens genau so gute Ergebnisse liefert. Es ist die Methode der „*Uncertainty Quantification*“ (UQ). Die Methode ist im

Buch von Xiu [2] beschrieben und wurde in der Fahrzeugdynamik zuerst von Fünfschilling und Perrin [3] und dann von Kewlani u.a. angewandt [4]. In [3] wird der Einfluss der Variationen der Rad/Schiene Kontaktgeometrie auf die Fahrzeugdynamik untersucht. In [4] untersucht Kewlani u.a. die senkrechte Dynamik eines 'Viertel-Wagen Modells' mit zwei Freiheitsgraden und parametrischer Unsicherheit unter einer deterministischen Erregung. In [4] findet man auch eine kurze Beschreibung der Methode.
In dieser Arbeit wollen wir statt der Abschätzung von Ergebnissen eines erregten Systems mit parametrischer Unsicherheit die Abschätzung der kritischen Parameterwerte, wie die kritische Geschwindigkeit eines Eisenbahnfahrzeugs, unter Einfluss der parametrischen Unsicherheit berechnen. Die Berechnungsmethode wird kurz präsentiert und auf einem einfachen Drehgestell Modell angewandt. Die Ergebnisse der Anwendung werden präsentiert und diskutiert. Zum Schluss wird die Erweiterung auf realistische Probleme mit mehreren Freiheitsgraden und einer hohen Zahl von Federn und Dämpfern in einem Modell eines Wagens erörtert. Dieses Problem ist, wie diese Präsentation, ein Teil der laufenden Doktorarbeit von Daniele Bigoni.

2. Die Methode der „Uncertainty Quantification“ (UQ)

Das fahrzeugdynamische Problem ist in der Form

$$\mathrm{d}\boldsymbol{q}/\mathrm{d}t = \mathbf{F}(\boldsymbol{q},\boldsymbol{Z}) \tag{1}$$

mit zugehörigen Bindungen und Anfangsbedingungen gegeben. Hier ist $\boldsymbol{q}(t, \boldsymbol{Z})$ ein 2N-dimensionaler Vektor, wo N die Zahl der Freiheitsgrade des dynamischen Systems ist, und t die Zeit ist. $\mathbf{F}$ ist eine Vektorfunktion und $\boldsymbol{Z}$ ist ein Vektor dessen M Komponenten Zufallsparameter mit gegebenen Verteilungen sind. Gesucht sind der Mittelwert $\boldsymbol{E}\,[\boldsymbol{q}(t, \boldsymbol{Z})]$ und der Varianz $\boldsymbol{V}\,[\boldsymbol{q}(t, \boldsymbol{Z})]$ für $t > 0$.
Die Lösung des dynamischen Problems mit Parameterverteilungen, die als die einzige Bedingung stetig sind, wird durch eine Reihenentwicklung der Verteilungen in orthogonale Funktionen angenähert, für die der Einfachheit wegen ein Basis von Hermitpolynomen $\{\mathcal{H}_k(\boldsymbol{Z})\}_{|k|\leq N}$ gewählt wird. Jedes Glied in der Entwicklung besteht aus einem Tensor Produkt von drei 'eindimensionalen' Hermitpolynomen. k ist ein dreidimensionales Multiindex, dem dreidimensionalen euklidischen Raum entsprechend – zum Beispiel, für $|k| = 0$ ist $\underline{k} = (0,0,0)$ und $k = 1$. Für $|k| = 1$ gibt es drei Kombinationen für $\underline{k}$: $(1,0,0) \sim k = 2$, $(0,1,0) \sim k = 3$ und $(0,0,1) \sim k = 4$ und so weiter. Für $|k| = 2$ gibt es z.B. sechs Kombinationen des dreidimensionalen Multiindexes. Ein Vorteil der Anwendung von Hermitpolynomen ist, dass in vielen Rechenprogrammen wie z.B. MATLAB eine Annäherung einer willkürlichen Funktion mittels Hermitpolynomen nach der Eingabe einiger Stützpunkte durch eine Operation durchführbar ist.
Im weiteren Verlauf kann man zwei Methoden anwenden. Erstens kann man eine Spektralmethode, die als 'Generalized Polynomial Chaos' oder gPC bekannt ist, oder die 'Stochastische Kollokation Methode' (SKM) anwenden. Der Name 'Generalized Polynomial Chaos' wurde von Wiener [5] 1938 eingeführt, und er hat nichts mit dem dynamischen Begriff 'Chaos' zu tun. In gPC werden die Reihenentwicklungen der Parameterverteilungen in das dynamische Problem (1) substituiert, und unter Ausnutzung der Ortogonalität der unterschiedlichen Hermitpolynome vereinfacht. Dadurch entsteht ein größeres dynamisches System, worin in jeder einzelnen Gleichung nur eine Kombination von Hermitpolynomen statt der ursprünglichen Verteilungsfunktionen auftritt. Die Gleichungssysteme sind in Gruppen aufgeteilt, die die unterschiedlichen Kombinationen von Hermitpolynomen enthalten und sonst alle gleich sind, was die numerische Lösung des größeren dynamischen Systems sehr vereinfacht.
Für die numerische Lösung $q_j(t_i)$ des 'großen Problems', müssen die Stützpunkte t_i für alle j, das heißt in allen Gruppen, gleich gewählt werden, damit man den Mittelwert und Varianz über j berechnen kann. Das Ergebnis liefert dann die Annäherung von dem gesuchten Mittelwert $\boldsymbol{E}\,[\boldsymbol{q}(t, \boldsymbol{Z})]$ und Varianz $\boldsymbol{V}\,[\boldsymbol{q}(t, \boldsymbol{Z})]$ für $t > 0$.

In SKM werden die Hermitpolynome durch Kollokation angenähert. Aus der Verteilungsfunktion wird eine Folge von Punkten - die sogenannten Kollokationspunkte – gewählt, deren Funktionswerte zur Bestimmung der Koeffizienten in der gewählten Entwicklung in Hermitpolynome dienen. Die dadurch gebildete Annäherung nennt man eine *Surrogatfunktion.* Dadurch entsteht wie bei der gPC Methode ein großes Gleichungssystem, das in Gruppen aufgeteilt ist. Auf die Wahl der Kollokationspunkte wird hier nicht eingegangen, aber es ist klar dass der Fehler in der Annäherung von dieser Wahl abhängig ist. Bis auf die unterschiedlichen Kombinationen von Hermitpolynomen, sind alle Gruppen gleich, und das dynamische Problem wird, wie oben für die gPC Methode beschrieben, numerisch gelöst. Für mehr Information wird der Leser auf die früher genannte Literatur [2] [4] hingewiesen.

3. Ein Beispiel aus der Eisenbahnfahrzeugdynamik

In diesem Beispiel wollen wir eine Simulation wie die von Mazzola und Bruni [1], Fünfschilling und Perrin [3] und Kewlani u.a. [4] nicht durchführen, sondern den wichtigen Systemparameter, *die kritische Geschwindigkeit eines Eisenbahnfahrzeugs* abschätzen. Als Modell wählen wir das früher untersuchte 'einfache Cooperrider Drehgestell' [6], *Bild 1*. Das Drehgestell Modell besteht aus zwei Radsätzen mit konischem Radprofil, die frei rotieren können aber sonst fest mit dem Drehgestell Rahmen verbunden sind. Sie werden alle als Festkörper betrachtet. Wir sind nur in den Querbewegungen interessiert und nehmen deswegen an, dass die senkrechten Bewegungen und Beschleunigungen so klein sind, dass sie in allen Bewegungsgleichungen vernachlässigbar sind. Deswegen werden die senkrechten Elemente der Aufhängung vernachlässigt, und die Verbindung zwischen dem Rahmen und dem Wagenkasten besteht nur aus einem Paar von seitlichen Feder-Dämpfern und einer Torsionsfeder, die alle lineare Abhängigkeiten besitzen.

Here please insert your fig 1 in WORD.docx

Bild 1. Das einfache Cooperrider Drehgestell Modell von oben

Das dynamische System ist [6]:

$$\begin{aligned} m\ddot{q}_1 &= -2D_2\dot{q}_1 - 2k_4q_1 - 2\left[F_x\left(\xi_{x_1},\xi_{y_1}\right) + F_x\left(\xi_{x_2},\xi_{y_2}\right)\right] - F_T(q_1 + haq_2) - F_T(q_1 - haq_2), \\ I\ddot{q}_2 &= -k_6q_2 - 2ha\left[F_x\left(\xi_{x_1},\xi_{y_1}\right) - F_x\left(\xi_{x_2},\xi_{y_2}\right)\right] - 2a\left[F_y\left(\xi_{x_1},\xi_{y_1}\right) + F_y\left(\xi_{x_2},\xi_{y_2}\right)\right] \end{aligned} \quad (2)$$

D_2 ist der lineare Dämpfungswert und k_4 und k_6 sind die linearen Federsteifigkeiten. F_x und F_y sind die Schlupfkräfte in bzw. Quer- und Längsrichtung. F_T ist die Rückführungskraft des Spurkranzes. Das Drehgestell läuft auf einem geraden und idealen Festkörpergleis mit konstantem Adhäsionsbeiwert.

$$F_x\left(\xi_x,\xi_y\right) = \frac{\xi_x}{\phi}\frac{F_R(\xi_x,\xi_y)}{\xi_R(\xi_x,\xi_y)} \quad , \quad F_y\left(\xi_x,\xi_y\right) = \frac{\xi_y}{\psi}\frac{F_R(\xi_x,\xi_y)}{\xi_R(\xi_x,\xi_y)} \ ,$$

$$\xi_R\left(\xi_x,\xi_y\right) = \sqrt{\frac{\xi_x^2}{\phi^2} + \frac{\xi_y^2}{\psi^2}} \ ,$$

$$\frac{F_R\left(\xi_x,\xi_y\right)}{\mu N} = \begin{cases} u(\xi_R) - \frac{1}{3}u^2(\xi_R) + \frac{1}{27}u^3(\xi_R) & \text{für } u(\xi_R) < 3 \\ 1 & \text{für } u(\xi_R) \geq 3 \end{cases} ,$$

$$u(\xi_R) = \frac{G\pi ab}{\mu N}\xi_R \ ,$$

Die Schlupfkräfte sind laut Vermeulen und Johnson [7]:

$$\xi_{x_1} = \frac{\dot{q}_1}{v} + ha\frac{\dot{q}_2}{v} - q_2 \ , \qquad \xi_{y_1} = a\frac{\dot{q}_2}{v} + \frac{\lambda}{r_0}(q_1 + haq_2) \ ,$$

$$\xi_{x_2} = \frac{\dot{q}_1}{v} - ha\frac{\dot{q}_2}{v} - q_2 \ , \qquad \xi_{y_2} = a\frac{\dot{q}_2}{v} + \frac{\lambda}{r_0}(q_1 - haq_2) \ .$$

Die Rückführungskraft des Spurkranzes ist angenähert durch eine nichtlineare Feder mit Spiel und hoher Steifigkeit:

$$F_T(x) = \begin{cases} \exp\big(-\alpha/(x - x_f)\big) - \beta x - \kappa \,, & 0 \le x < b \\ k_0 \cdot (x - \delta) \,, & b \le x \\ -F_T(-x) \,, & x < 0 \end{cases} ,$$

Die angewandten Parameterwerte sind unten aufgelistet:

$m = 4963\ kg$	$h = 1.5\ m$	$a = 0.7163\ m$
$I = 8135\ kg \cdot m^2$	$D_2 = 29200\ N \cdot s/m$	$k_0 = 14.60 \cdot 10^6\ N/m$
$k_4 = 0.1823 \cdot 10^6\ N/m$	$k_6 = 2.710 \cdot 10^6\ N/m$	$\lambda = 0.05$
$r_0 = 0.4572\ m$	$b = 0.910685 \cdot 10^{-2}\ m$	$\phi = 0.60252$
$\psi = 0.54219$	$G\pi ab = 6.563 \cdot 10^6\ N$	$\mu N = 10^4\ N$
$\delta = 0.0091\ m$	$\alpha = 0{,}1474128791 \cdot 10^{-3}$	$\beta = 1{,}016261260$
$\kappa = 1{,}793756792$	$x_f = 0.9138788366 \cdot 10^{-2}$	

3.1 Die kritische Geschwindigkeit des deterministischen Modells

Wie erwartet ist die kritische Geschwindigkeit des Drehgestell Modells die höchste Geschwindigkeit an der die stationäre Gleichgewichtslösung des dynamischen Problems eindeutig ist. Im Zustands-Parameterraum haben wir also mit einem Problem zu tun, das dieselben Eigenschaften hat wie das im *Bild 2*.

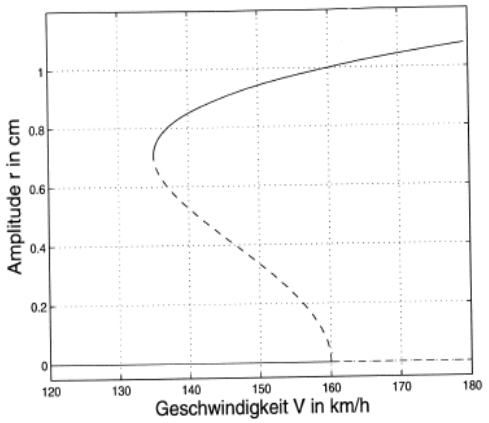


Bild 2. Beispiel eines Verzweigungsdiagrammes für ein Drehgestell, Amplitude der Querbewegung des führenden Radsatzes vs. Fahrzeuggeschwindigkeit. Der ungestörte Lauf ist die Nulllösung – die Abszissenachse. Der kleinste Verzweigungspunkt, liegt bei 134 km/h aber nicht auf dieser Achse. Dort verzweigen eine instabile – gestrichelt – und eine stabile – aufgezogen – periodische Lösung, die die Schlingerbewegung darstellt, von einander. Die Nulllösung ist für V > 160 km/h instabil.

Die subkritische Verzweigung der instabilen, periodischen Lösung liegt in unserem Fall bei v_L = 66.6107 m/s ~ 239.80 km/h. Über v_L ist unsere stationäre Gleichgewichtslösung instabil. v_L wurde mit großer Genauigkeit durch eine Stabilitätsuntersuchung der stationären Lösung unter Anwendung der Eigenwerte der Jacobiante des um Null linearisierten Systems, gefunden. Danach wurde die stabil schwingende Bewegung – die Schlingerbewegung – im Zustandsraum gefunden,

und die kritische Geschwindigkeit wurde dann durch *Ramping* dieser Lösung berechnet. Bei der Ramping wird die Geschwindigkeit als eine stetige und langsam abnehmende Funktion der Zeit ins dynamische System eingeführt. [8]:

$$\dot{v} = \begin{cases} 0\,, & \text{wenn } t < t_{st} \vee \|\vec{q}\|_2 < \epsilon_{min} \\ -\Delta\,, & \text{sonst} \end{cases} \tag{3}$$

Hierdurch wurde die kritische Geschwindigkeit des deterministischen Modells, v_{NL} = 62.0206 m/s ~ 223.27 km/h berechnet. Die Methode und das Ergebnis sind im *Bild 3* dargestellt.

Here please insert your picture fig. 2 in WORD with German text
Lateral Amplitude → Amplitude der Querschwingung
Non-Lin. Crit Speed → Kritische Geschw. Lin. Crit. Speed → Stabilitätsgrenze
Hopf-Bifurkation → Hopf Verzweigung Ramping method → Ramping Methode
Yaw amplitude → Amplitude des Drehwinkels Speed → Geschwindigkeit

Bild 3. Die nichtlineare Dynamik des deterministischen Systems. Die Nulllösung ist über v_L = 66.6107 m/s instabil, und die kritische Geschwindigkeit v_{NL} = 62.0206 m/s ist durch Ramping berechnet

3.2 Die kritische Geschwindigkeit des stochastischen Modells

Es wird angenommen dass die Parameter der Elemente der Aufhängung mit einer Liefertoleranz behaftet sind. Wir nehmen an dass die Unsicherheit durch eine Gauß Verteilung mit dem nominellen Wert als Mittelwert und einer definierten Varianz gegeben ist. Dadurch wird unser deterministisches Modell in ein stochastisches Modell verwandelt, wo jede einzelne Lösung eine partikulare Realisierung darstellt, und probabilistische Momente die Statistik der stochastischen Lösung beschreibt. Die Parameter sei gegeben durch:

$$\begin{aligned} k_6 &\sim \mathcal{N}\left(\mu_{k_6}, \sigma_{k_6}^2\right) = \mathcal{N}\left(2.71 \cdot 10^6 \, \frac{N}{m}, 1.84 \cdot 10^{10} \left(\frac{N}{m}\right)^2\right), \quad (\text{StA.} \sim 5\%) \\ k_4 &\sim \mathcal{N}\left(\mu_{k_4}, \sigma_{k_4}^2\right) = \mathcal{N}\left(9.12 \cdot 10^4 \, \frac{N}{m}, 4.15 \cdot 10^7 \left(\frac{N}{m}\right)^2\right), \quad (\text{StA.} \sim 7\%) \\ D_2 &\sim \mathcal{N}\left(\mu_{D_2}, \sigma_{D_2}^2\right) = \mathcal{N}\left(1.46 \cdot 10^4 \, N \cdot \frac{s}{m}, 1.07 \cdot 10^6 \left(N \cdot \frac{s}{m}\right)^2\right). \quad (\text{StA.} \sim 7\%) \end{aligned} \tag{4}$$

Std. of approx. → Standardabweichung von ungefähr

Wir anwenden die 'Stochastische Kollokation Methode' (SKM) aus Abschnitt 2 auf das dynamische Problem (2). Jeder der dadurch entstandenen Differentialgleichungssysteme wird wie im Abschnitt 3.1 als ein deterministisches Problem gelöst. Von allen dadurch gefundenen kritischen Geschwindigkeiten werden Mittelwert und Varianz berechnet.
Wir fangen mit einem einfachen eindimensionalen Fall an. In der Mathematik wird er Codimensionn 1 genannt, weil die Dimensionszahl sich auf die Anzahl der Parameter, zum Unterschied von der Dimension des dynamischen Systems (2), bezieht. Die Dimensionszahl ist hier 4, weil das System (2) durch die Einführung neuer Variable: $x_1 = q_1, x_2 = \dot{q_1}, x_3 = q_2$ und $x_4 = \dot{q_2}$ für die Lösung in vier Differentialgleichungen erster Ordnung für die neuen Variable x_1 - x_4 überführt wird.
Die Lösung des stochastischen Problems mit SKM wird den Lösungen desselben Problems mit sowohl der Monte-Carlo Methode (MC) wie der Quasi-Monte-Carlo Methode (QMC), die schneller arbeitet, gegenüber gestellt um die Genauigkeit und die Rechenzeit der Methoden zu vergleichen. Beide Methoden arbeiten gut, solange die Lösung des Problems hinreichend glatt im Zufallsraum ist. Es ist mit Sicherheit nicht der Fall, wenn Verzweigungen im dynamischen Problem existieren, und unterschiedliche Realisierungen der Zufallsparameter zu unterschiedlichen Attraktoren in dem Zustands-Parameterraum führen können.
Zunächst wird ein Fall mit Codimensionn 1 Unsicherheit untersucht. Die Federsteifigkeit k_4 sei

durch eine Normalverteilung gegeben, und die berechneten Mittelwert und Varianz der kritischen Geschwindigkeit sind im *Bild 4* abgebildet.

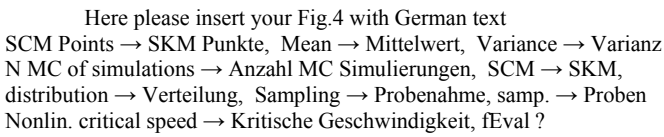


Bild 4. Die Ergebnisse der Berechnung der kritischen Geschwindigkeit mit Hilfe sowohl der SKM wie der MC und QMC Methoden. Nur k_4 ist mit Unsicherheit behaftet. Links die Abschätzung des Mittelwerts und der Varianz. Rechts Histogramme der kritischen Geschwindigkeit. Für die MC Methode wurden 500 Simulationen des Modells und für die SKM Methode nur zwei Funktionsauswertungen benutzt. Für die Verteilungsfunktion der SKM Methode wurden 10^5 Realisierungen der Surrogatfunktion angewandt. Die Standard Abweichung ist beschattet – blau für SKM und rot für MC.

Man sieht dass die Abschätzung bei SKM schon bei niedriger Ordnung zufriedenstellend ist, und der Gewinn bei höhrerer Ordnung nur gering ist. Es bedeutet dass die ersten wenigen Glieder in der Entwicklung laut Abschnitt 2 für eine gute Abschätzung der Verteilung der kritischen Geschwindigkeit ausreichen.
Bild 5 ist wie *Bild 4*, bloß ist statt k_4 hier k_6 durch eine Normalverteilung gegeben. Wieder reichen die ersten wenigen Glieder in der Entwicklung laut Abschnitt 2 für eine gute Abschätzung der Verteilung der kritischen Geschwindigkeit aus. Ferner sieht man dass die Steifigkeit der Torsionsfeder einen größeren Einfluss als die Steifigkeit der Querfederung auf die kritische Geschwindigkeit ausübt.

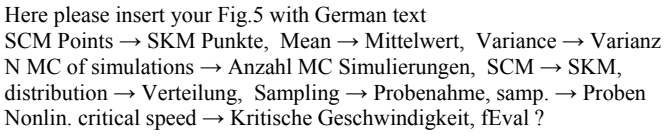


Bild 5. Die Ergebnisse der Berechnung der kritischen Geschwindigkeit mit Hilfe sowohl der SKM wie der MC und QMC Methoden. Nur k_6 ist mit Unsicherheit behaftet sonst wie *Bild 4.* Die Standard Abweichung ist beschattet – blau für SKM und rot für MC.

Schließlich wird die SKM Methode auf das Problem (1) mit den drei Verteilungen (3) angewandt. Die Ergebnisse sind im *Bild 6* präsentiert. Wieder sieht man dass eine Annäherung niedriger Ordnung für die Berechnung der Lösung mit der größten Genauigkeit ausreicht.

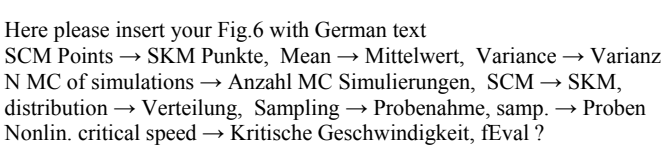


Bild 6. Die Ergebnisse der Berechnung der kritischen Geschwindigkeit mit Hilfe sowohl der SKM wie der MC und QMC Methoden mit dreidimensionaler Unsicherheit. Die Anzahl der Auswertungen bei MC wurde hier auf 10^3 erhöht, sonst wie *Bilder 4 und 5*. Rechts Histogramme der kritischen Geschwindigkeit. Die Standard Abweichung ist beschattet – blau für SKM und rot für MC.

Tabelle 1 zeigt das Endergebnis mit der größten erreichten Genauigkeit bei der Anwendung der drei Berechnungsmethoden: Monte-Carlo (MC), Quasi-Monte-Carlo (QMC) und Stochastische

Kollokation (SKM). Es ist interessant zu bemerken dass die Varianzen in den zwei- und dreidimensionalen Fällen fast gleich der Summe der Varianzen der eindimensionalen Fälle sind. Es bedeutet dass die gegenseitige Beeinflußung der drei Elemente der Aufhängung durch die Nichtlinearität des Problems bei den gewählten Varianzen unbedeutend ist.

	MC				**QMC**				**SKM**			
	$\boldsymbol{\mu}$	$\boldsymbol{\sigma^2}$	**#fA**	**CPUt**	$\boldsymbol{\mu}$	$\boldsymbol{\sigma^2}$	**#fA**	**CPUt**	$\boldsymbol{\mu}$	$\boldsymbol{\sigma^2}$	**#fA**	**CPUt**
k_6	62,26	1,64	169	~24S	62,24	1,47	152	~21 S	62,23	1,55	2	~10M
k_4	62,23	0,14	17	~2,5S	62,25	0,14	22	~3 S	62,25	0,14	2	~11M
D_2	62,23	0,02	9	~1 S	62,25	0,02	4	~30M	62,25	0,03	2	~11M
k_6, k_4	62,22	1,53	148	~21 S	62,22	1,62	152	~22 S	62,28	1,69	4	~36M
k_6, D_2	62,18	1,72	216	~30 S	62,24	1,50	142	~20 S	62,28	1,57	4	~37M
k_4, D_2	62,25	0,17	25	~3,5S	62,25	0,16	25	~3,5S	62,30	0,17	4	~35M
k_6, k_4, D_2[1]	62,18	1,68	221	~32 S	62,23	1,63	154	~22 S	62,23	1,72	8	~1 S

Here please insert your table 1 with German text
eval → Ausw., max. order → Max. Ordnung

Tabelle 1. Die geschätzten Mittelwerte und Varianzen der krirtischen Geschwindigkeit bei der Anwendung der MC, QMC und SKM Methoden.

4. Diskussion und Ausblick

In dieser Arbeit haben wir gezeigt, wie die Fertigungstoleranzen in dynamische Untersuchungen von Fahrzeugen eingeführt werden können. Ein neues Verfahren, die stochastische Kollokation als 'Uncertainty Quantification', wird angewandt, und die Genauigkeit und der Rechenaufwand mit denen der Anwendung des Monte-Carlo Verfahrens verglichen. Die 'Uncertainty Quantification' wird zur Abschätzung der berechneten kritischen Geschwindigkeit angewandt, und die kritische Geschwindigkeit wird als ein Mittelwert mit Varianz geliefert. Die Ergebnisse zeigen dass unter der Voraussetzung derselben Genauigkeit ist der Konvergenz des neuen Verfahrens dem Konvergenz des Monte-Carlo Verfahrens überlegen. Rechenaufwand Der gesamte Rechenaufwand ist selbstverständlich größer als der einer deterministischen Berechnung, weil dasselbe dynamische System, bloß mit unterschiedlichen Parameterwerten, wiederholt numerisch gelöst werden muss. Unter diesen Umständen lässt sich aber die Rechenzeit bei einer geschickten Anwendung der Parallelisierung erheblich reduzieren. Die Dynamik des Fahrzeugmodells wird unterwegs berechnet, weil sie für die Berechnung der kritischen Geschwindigkeit die Grundlage bildet, aber das Ergebnis wird hier wegen der Begrenzung der Länge dieser Veröffentlichung nicht präsentiert. Wir haben ein sehr einfaches Beispiel gewählt um die Überlegenheit der Methode gegenüber dem MC Verfahren zu demonstrieren. Deswegen haben wir durch die Anwendung nur einer Verteilung für die lateralen Feder- bzw. Dämpferkräfte den Einfluß der in einem realistischen Wagen fehlenden Symmetrie hier nicht untersucht. Der Rechenaufwand wächst bei der Untersuchung eines ganzen realistischen Fahrzeugs mit einer hohen Codimension, die leicht 20 übersteigt, gewaltig. Deswegen wird die Doktorarbeit mit dem Ziel den Rechenaufwand zu reduzieren weitergeführt. Die Codimension könnte durch Anwendung statistischer Methoden zur Auswahl der einflußreichsten Systemparameter reduziert werden.

5. Literatur

[1] *Mazzola, L. Bruni, S.*: Effect of Suspension Parameter Uncertainty on the Dynamic Behavior of Railway Vehicles, Applied Mechanics and Materials (2011) 104, S. 177-185.

[2] *Xiu, D.*: Numerical Methods for Stochastic Computations: A Spectral Method Approach,

[1] For the full 3D uncertainty problem, the number of evaluation used for MC has been increased to 10^3.

Princeton University Press.
[3] *Fünfschilling, C., Perrin, G., Kraft, S.*: Propagation of Variability in Railway Dynamic Simulations: Application to Virtual Homologation (2012) Vehicle System Dynamics, 50:sup 1, S. 245-261, Taylor & Francis.
[4] *Kewlani, G., Crawford, J., Iagnemma, K.*: A polynomial chaos approach to the analysis of vehicle dyanmics under uncertainty, Vehicle System Dynamics (2012) 50, 5, S. 749-774, Taylor & Francis.
[5] *Wiener, N.*: The homogeneous Chaos, American Journal of Mathematics (1938) 60, 4, S. 897-936.
[6] *True, H.; Kaas-Petersen, Chr.*: A Bifurcation Analysis of Nonlinear Oscillations in Railway Vehicles, Proc. 8th IAVSD-IUTAM Symposium on Vehicle System Dynamics, The Dynamics of Vehicles on Roads and Tracks (1984) S. 655 – 665, Swets & Zeitlinger, Lisse, NL.
[7] *Vermeulen, P.J., Johnson, K.L.*: Contact of nonshperical elastic bodies transmitting tangential forces, Journal of Applied Mathematics (1964) 31, S. 338-340.
[8] *True, H.*: Die Berechnung der kritischen Geschwindigkeit eines Eisenbahnfahrzeuges: Die Richtige, die Falsche und die Zufallsmethode, ZEV Glasers Annalen, 135, Tagungsband SFT Graz 2011, S. 162—169.

SENSITIVITY ANALYSIS OF THE CRITICAL SPEED IN RAILWAY VEHICLE DYNAMICS

Daniele Bigoni, Hans True, Allan P. Engsig-Karup
Department of Applied Mathematics and Computer Science, Technical University of Denmark
Matematiktorvet, building 303B
DK-2800 Kgs. Lyngby, Denmark
E-mail: dabi@dtu.dk

Abstract

We present an approach to global sensitivity analysis aiming at the reduction of its computational cost without compromising the results. The method is based on sampling methods, cubature rules, High-Dimensional Model Representation and Total Sensitivity Indices. The approach has a general applicability in many engineering fields and does not require the knowledge of the particular solver of the dynamical system. This analysis can be used as part of the virtual homologation procedure and to help engineers during the design phase of complex systems.
The method is applied to a half car with a two-axle Cooperrider bogie, in order to study the sensitivity of the critical speed with respect to suspension parameters. The importance of a certain suspension component is expressed by the variance in critical speed that is ascribable to it. This proves to be useful in the identification of parameters for which the exactness of their values is critically important.

1. INTRODUCTION

The past couple of decades have seen the advent of computer simulations for the study of deterministic dynamical systems arising from any field of engineering. The reasons behind this trend are both the enhanced design capabilities during production and the possibility of understanding dangerous phenomena. However, deterministic dynamical systems fall short in the task of giving a complete picture of reality: several sources of uncertainty can be present when the system is designed and thus obtained results refer to single realizations, that in a probabilistic sense have measure zero, i.e. they never happen in reality. The usefulness of these simulations is however proved by the achievements in Computer-Aided Design (CAD).
The studies of stochastic dynamical systems allow for a wider analysis of phenomena: deterministic systems can be extended with prior knowledge on uncertainties with which the systems are described. This enables an enhanced analysis and can be used for risk assessment subject to such uncertainties and is useful for decision making in the design phase.
In the railway industry, stochastic dynamical systems are being considered in order to include their analysis as a part of the virtual homologation procedure [1], by means of the framework for global parametric uncertainty analysis proposed by the OpenTURNS consortium. This framework splits the uncertainty analysis task in four steps:

A. Deterministic modeling and identification of Quantities of Interest (QoI) and source of uncertainties
B. Quantification of uncertainty sources by means of probability distributions
C. Uncertainty propagation through the system
D. Sensitivity analysis

Railway vehicle dynamics can include a wide range of uncertainty sources. Suspension characteristics are only known within a certain tolerance when they exit the manufacturing factory and are subject to wear over time that can be described stochastically. Other quantities that are subject to uncertainties are the mass and inertia of the bodies, e.g. we don't know exactly how the wagon will be loaded, the wheel and track geometries, that are subject to wear over time, and also external loadings like wind gusts.
In this work the QoI will be the critical speed of a fixed half-wagon with respect to uncertain suspension components (step A). The deterministic and stochastic models will be presented in section 2. Step B requires measurements of the input uncertainty that are not available to the authors, so the probability distribution of the suspension components will be assumed to be Gaussian, without losing the generality of application of the methods used in C and D.
Techniques for Uncertainty Quantification (UQ) will be presented in section 3.1. They have already been applied in [2] and [3] to perform an analysis of Uncertainty propagation (step C). They will turn useful also in section 3.2 and 3.3 for the sensitivity analysis technique to be presented (step D). This is based on Total Sensitivity Indices (TSI) obtained from the ANOVA expansion of the function associated to the QoI [4]. Section 4 will contain the results of such analysis.

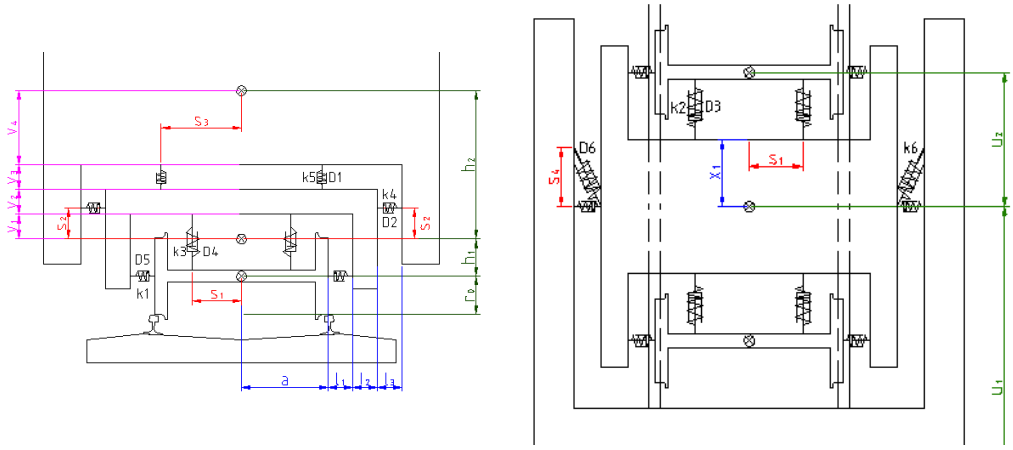


Fig. 1 The half-wagon equipped with the Cooperrider bogie.

2. THE VEHICLE MODEL

In this work we will consider a fixed half wagon equipped with a Cooperrider bogie, running on tangent track with wheel profile S1002 and rail UIC60. The position of the suspension components is shown in Fig. 1. In [5] a framework for the simulation of the dynamics of complete wagons running on straight and curved tracks has been implemented and tested based on the Newton-Euler formulation of the dynamical system:

$$\sum_{i=1}^{n} \vec{F}_i = m\vec{a}\,, \qquad \sum_{i=1}^{m} M_i = \frac{d}{dt}\left([J]\cdot\vec{\omega}\right) + \vec{\omega}\times\left([J]\cdot\vec{\omega}\right), \tag{1}$$

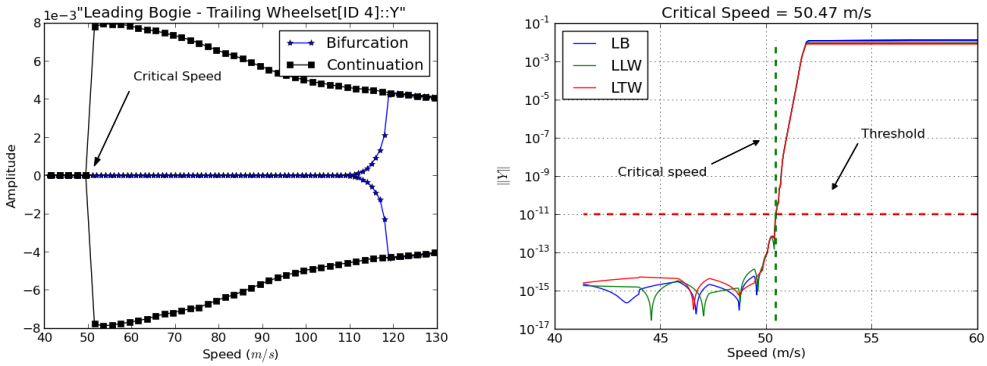
where F_i and M_i are respectively the forces and the torques acting on the bodies, m and $[J]$ are the mass and inertia of the bodies, $\vec{a}$ and $\dot{\vec{\omega}}$ are the acceleration and the angular acceleration of the bodies.
In this work the wagon will be fixed in order to alleviate the lateral oscillations during the hunting motion that would, in some cases, break the computations. The mathematical analysis and the generality of the methods proposed are not weakened by this assumption, even if the results may change for different settings. The wheel-rail interaction is modeled using tabulated values generated with the routine RSGEO [6] for the static penetration at the contact points. These values are then updated using Kalker's work [7] for the additional penetrations. The creep forces are approximated using Shen-Hedrick-Elkins nonlinear theory [8]. The complete deterministic system

$$\frac{d}{dt}\mathbf{u}(t) = \mathbf{f}(\mathbf{u}, t)\,, \tag{2}$$

is nonlinear, non-smooth, and it has 28 degrees of freedom.

2.1 Nonlinear dynamics of the deterministic model

The deterministic dynamics of the complete wagon with a couple of Cooperrider bogies were analyzed in [5]. The stability of the half-wagon model considered in this work is characterized by a subcritical Hopf-bifurcation at $v_L = 114\ m/s$, as it is shown in Fig. 2a, and a critical speed $v_{NL} = 50.47\ m/s$. The critical speed is found using a continuation method from the periodic limit cycle detected at a speed greater than the Hopf-bifurcation speed v_L. In order to save computational time, we try to detect the periodic limit cycle at speeds lower than v_L perturbing the system as described in [9]. This is the approach that we will take during all the computations of critical speeds in the next sections. The criterion used in order to detect the value of the critical speed is based on the power of the lateral oscillations in a $1s$ sliding window of the computed solution. Fig. 2b shows how this criterion is applied.



(a) bifurcation diagram *(b) Critical speed detection criteria*

Fig. 2 Left: complete bifurcation diagram where the folding point is detected by continuation (ramping) method from the periodic limit cycle. Right: criterion for the determination of the critical speed based on the power of the lateral oscillations in a sliding window. LB, LLW and LTW stand for the bogie frame, the leading wheel set and the trailing wheel set respectively.

2.2 The stochastic model

In the following we will assume that the suspension characteristics are not deterministically known. Rather, they are described by probability distributions stemming from the manufacturing uncertainty or the wear.
If experimental information is available, then some standard distributions can be assumed and an optimization problem can be solved in order to determine the statistical parameters of such distributions (e.g. mean, variance, etc.). Alternatively the probability density function of the probability distribution can be estimated by Kernel smoothing [10].
Due to the lack of data to the authors, in this work the probability distributions associated with the suspension components will be assumed to be Gaussian around their nominal value, with a standard deviation of 5%. We define $\mathbf{Z}$ to be the *d*-dimensional vector of random variables $\{z_i \sim N(\mu_i, \sigma_i)\}_{i=1}^{d}$ describing the distributions of the suspension components, where *d* is called the co-dimension of the system. The stochastic dynamical system is then described by

$$\frac{d}{dt}\mathbf{u}(t,\mathbf{Z}) = \mathbf{f}(\mathbf{u},t,\mathbf{Z}), \qquad (0,T]\times\mathbf{R}^d . \tag{3}$$

3. SENSITIVITY ANALYSIS

Sensitivity analysis is used to describe how the model output depends on the input parameters. Such analysis enables the user to identify the most important parameters for the model output. Sensitivity analysis can be viewed as the search for the direction in the parameter space with the fastest growing perturbation from the nominal output.
One approach of sensitivity analysis is to investigate the partial derivatives of the output function with respect to the parameters in the vicinity of the nominal output. This approach goes by the name of local sensitivity analysis, stressing the fact that it works only for small perturbations of the system.
When statistical information regarding the parameters is known, it can be embedded in the global sensitivity analysis, which is not restricted to small perturbations of the system, but can handle bigger variability in the parameter space. This is the focus of this work and will be described in the following sections.

3.1 Uncertainty Quantification

The solution of (3) is $\mathbf{u}(t,\mathbf{Z})$, varying in the parameter space. In uncertainty quantification we are interested in computing the density function of the solution and/or its first moments, e.g. mean and variance:

$$\begin{aligned} \mu_{\mathbf{u}}(t) &= \mathbf{E}\left[\mathbf{u}(t,\mathbf{Z})\right]_{\rho_{\mathbf{Z}}} = \int_{\Omega^d} \mathbf{u}(t,\mathbf{z}) dF_{\mathbf{Z}}(\mathbf{z}), \\ \sigma^2_{\mathbf{u}}(t) &= \mathbf{V}\left[\mathbf{u}(t,\mathbf{Z})\right]_{\rho_{\mathbf{Z}}} = \int_{\Omega^d} \left(\mathbf{u}(t,\mathbf{z}) - \mu_{\mathbf{u}}(t)\right)^2 dF_{\mathbf{Z}}(\mathbf{z}), \end{aligned} \tag{4}$$

where $\rho_{\mathbf{Z}}(\mathbf{z})$ and $F_{\mathbf{Z}}(\mathbf{z})$ are the probability density function (PDF) and the cumulative distribution function (CDF) respectively. Several techniques are available to approximate these high-dimensional integrals. In the following we present the two main classes of these methods.

Sampling based methods

The most known sampling method is the Monte Carlo (MC) method, which is based on the law of large numbers. Its estimates are:

$$\begin{aligned} \mu_{\mathbf{u}}(t) &\approx \overline{\mu}_{\mathbf{u}}(t) = \frac{1}{M}\sum_{j=1}^{M}\mathbf{u}(t,\mathbf{Z}^{(j)}), \\ \sigma_{\mathbf{u}}^2(t) &\approx \overline{\sigma}_{\mathbf{u}}^2(t) = \frac{1}{M-1}\sum_{j=1}^{M}\left(\mathbf{u}(t,\mathbf{Z}^{(j)}) - \overline{\mu}_{\mathbf{u}}(t)\right)^2, \end{aligned} \tag{5}$$

where $\left\{\mathbf{Z}^{(j)}\right\}_{j=1}^{M}$ are realizations sampled randomly within the probability distribution of $\mathbf{Z}$. The MC method has a probabilistic error of $O\left(1/\sqrt{M}\right)$, thus it suffers from the work effort required to compute accurate estimates. However the MC method is very robust because this convergence rate is independent of the co-dimensionality of the problem, so it's useful to get approximate estimates of very high-dimensional integrals.
Sampling methods with improved convergence rates have been developed, such as Latin Hypercube sampling and Quasi-MC methods. However, the improved convergence rate comes at the expense of several drawbacks, e.g., the convergence of Quasi-MC methods is dependent of the co-dimensionality of the problem and Latin Hypercube cannot be used for incremental sampling.

Cubature rules

The integrals in (4) can also be computed using cubature rules. These rules are based on a polynomial approximation of the target function, i.e. the function describing the relation between parameters and QoI, so they have superlinear convergence rate on the set of smooth functions. Their applicability is however limited to low-co-dimensional problems because cubature rules based on a tensor grid suffer the curse of dimensionality, i.e. if m is the number of points used in the one dimensional rule and d the dimension of the integral, the number of points at which to evaluate the function grow as $O(m^d)$. They will however be presented here because they represent a fundamental tool for the creation of high-dimensional model representations that will be presented in the next section.
Let $\mathbf{Z}$ be a vector of independent random variables in the probability space $(\mathbf{D},\mathbf{B},F_{\mathbf{Z}})$, where $\mathbf{D}\subseteq\mathbf{R}^d$, $\mathbf{B}$ is the Borel set constructed on $\mathbf{D}$ and $F_{\mathbf{Z}}$ is a probability measure (i.e. the CDF of $\mathbf{Z}$). For this probability measure we can construct orthogonal polynomials $\left\{\phi_n(z_i)\right\}_{n=1}^{N_i}$ for $i=1\ldots d$, that form a basis for each independent dimension of $\mathbf{D}$ [11]. The tensor product of such a basis forms a basis for $\mathbf{D}$. From these orthogonal polynomials, the Gauss quadrature points and weights $\left\{\mathbf{z}_{j_i,\ldots,j_d},\mathbf{w}_{j_i,\ldots,j_d}\right\}_{j_i,\ldots,j_d=1}^{N_1,\ldots N_d}$ can be derived using the Golub-Welsch algorithm [11], obtaining approximations for (4):

$$\begin{aligned} \mu_{\mathbf{u}}(t) &\approx \overline{\mu}_{\mathbf{u}}(t) = \sum_{j_i=1}^{N_i}\cdots\sum_{j_d}^{N_d}\mathbf{u}\left(t,\mathbf{z}_{j_i,\ldots,j_d}\right)\mathbf{w}_{j_i,\ldots,j_d}, \\ \sigma_{\mathbf{u}}^2(t) &\approx \overline{\sigma}_{\mathbf{u}}^2(t) = \sum_{j_i=1}^{N_i}\cdots\sum_{j_d}^{N_d}\left(\mathbf{u}\left(t,\mathbf{z}_{j_i,\ldots,j_d}\right) - \overline{\mu}_{\mathbf{u}}(t)\right)^2\mathbf{w}_{j_i,\ldots,j_d}. \end{aligned} \tag{6}$$

Gauss quadrature rules of order N are accurate for polynomials of order up to degree $2N-1$. This high accuracy comes at the expense of the curse of dimensionality due to the use of tensor products in high-dimensional integration. This effect can be alleviated by the use of Sparse Grids techniques proposed by Smolyak [12] that use an incomplete version of the tensor product. However, in the following section we will see that we can often avoid working in very high-dimensional spaces.

3.2 High-dimensional model representations

High-dimensional models are very common in practical applications, where a number of parameters influence the dynamical behaviors of a system. These models are very difficult to handle, in particular if we consider them as black-boxes where we are only allowed to change parameters. One method to circumvent these difficulties is the

HDMR expansion [13], where the high-dimensional function $f:\mathbf{D}\to\mathbf{R}$, $\mathbf{D}\subseteq\mathbf{R}^n$ is represented by a function decomposed with lower order interactions:

$$f(x)\equiv f_0+\sum_i f_i(x_i)+\sum_{i<j} f_{ij}(x_i,x_j)+\cdots+f_{1,2,\cdots,n}(x_1,x_2,\ldots,x_n)\,. \tag{7}$$

This expansion is exact and exists for any integrable and measurable function f, but it is not unique. There is a rich variety of such expansions depending on the projection operator used to construct them. The most used in statistics is the ANOVA-HDMR where the low dimensional functions are defined by

$$\begin{aligned}
f_0^A(x)&\equiv P_0^A f(x)=\int_{\mathbf{D}} f(x)d\mu(x)\,,\\
f_i^A(x_i)&\equiv P_i^A f(x)=\int_{\mathbf{D}_i} f(x)\prod_{i\neq j} d\mu_j(x_j)-P_0^A f(x)\,,\\
f_{i_1\ldots i_l}^A(x_{i_1},\ldots,x_{x_l})&\equiv P_{i_1\ldots i_l}^A f(x)=\int_{\mathbf{D}_{i_1\ldots i_l}} f(x)\prod_{k\in\{i_1\ldots i_l\}} d\mu_k(x_k)-\sum_{j_1<\cdots<j_{l-1}\in\{i_1\ldots i_l\}} P_{j_1\ldots j_{l-1}}^A f(x)\\
&\quad-\ldots-\sum_{j\in\{i_1\ldots i_l\}} P_j^A f(x)-P_0^A f(x)\,,
\end{aligned} \tag{8}$$

where $\mathbf{D}_{i_1\ldots i_l}\subset\mathbf{D}$ is the hypercube excluding indices $i_1,\ldots,i_l$ and μ is the product measure $\mu(x)=\prod_i \mu_i(x_i)$. This expansion can be used to express the total variance of f, by noting that

$$\begin{aligned}
D&\equiv\mathbf{E}\left[f-f_0\right]^2=\sum_i D_i+\sum_{i<j} D_{ij}+\cdots+D_{1,2,\ldots,n}\,,\\
D_{i_1\ldots i_l}&=\int_{\mathbf{D}^{i_1\ldots i_l}}\left(f_{i_1\ldots i_l}^A\right)^2\prod_{k\in\{i_1\ldots i_l\}} d\mu_k(x_k)\,.
\end{aligned} \tag{9}$$

However, the high-dimensional integrals in the ANOVA-HDMR expansion are computationally expensive to evaluate.

An alternative expansion is the cut-HDMR, that is built by superposition of hyperplanes passing through the cut center $y=(y_1,\ldots,y_n)$:

$$\begin{aligned}
f_0^C(x)&\equiv P_0^C f(x)=f(y)\,,\\
f_i^C(x_i)&\equiv P_i^C f(x)=f^i(x_i)-P_0^C f(x)\,,\\
f_{i_1\ldots i_l}^C(x_{i_1},\ldots,x_{x_l})&\equiv P_{i_1\ldots i_l}^C f(x)=f^{i_1,\ldots,i_l}(x_{i_1},\ldots,x_{x_l})-\sum_{j_1<\cdots<j_{l-1}\in\{i_1\ldots i_l\}} P_{j_1\ldots j_{l-1}}^C f(x)\\
&\quad-\ldots-\sum_{j\in\{i_1\ldots i_l\}} P_j^C f(x)-P_0^C f(x)\,,
\end{aligned} \tag{10}$$

where $f^{i_1,\ldots,i_l}(x_{i_1},\ldots,x_{x_l})$ is the function $f(x)$ with all the remaining variables set to y. This expansion requires the evaluation of the function f on lines, planes and hyperplanes passing through the cut center.

If cut-HDMR is a good approximation of f at order L, i.e. considering up to L-terms interactions in (7), such expansion can be used for the computation of ANOVA-HDMR in place of the original function. This reduces the computational cost dramatically: let n be the number of parameters and s the number of samples taken along each direction (being them MC samples or cubature points), then the cost of constructing cut-HDMR is

$$\sum_{i=0}^{L}\frac{n!}{(n-i)!i!}(s-1)^i\,. \tag{11}$$

3.3 Total Sensitivity Index

The main task of Sensitivity Analysis is to quantify the sensitivity of the output with respect to the input. In particular it's important to know how much of this sensitivity is accountable to a particular parameter. With the focus on global sensitivity analysis, the sensitivity of the system to a particular parameter can be expressed by the variance of the output associated to that particular input.

One approach to this question is to consider each parameter separately and to apply one of the UQ techniques introduced in section 3.1. This approach goes by the name of one-at-a-time analysis. This technique is useful to get a first overview of the system. However, this technique lacks an analysis of the interaction between input

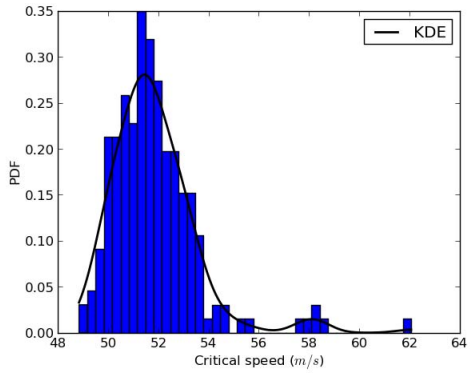


Fig. 3: Histogram of the Critical Speed obtained using Latin Hypercube sampling and the estimated density function (KDE) obtained using Kernel Smoothing.

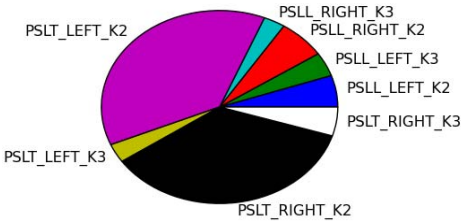


Fig. 4: Pie plot of the Total Sensitivity Indices on the reduced stochastic model, where only the most influential components are analyzed. (See Table 1 for an explanation of the notation used)

parameters, which in many cases is important.

A better analysis can be achieved using the method of Sobol' [14]. Here single sensitivity measures are given by

$$S_{i_1,\ldots,i_l} = \frac{D_{i_1,\ldots,i_l}}{D} \qquad \text{for } 1 \le i_1 < \cdots < i_l \le n\,, \tag{12}$$

where D and $D_{i_1,\ldots,i_l}$ are defined according to (9). These express the amount of total variance that is accountable to a particular combination $i_1,\ldots,i_l$ of parameters. The Total Sensitivity Index (TSI) is the total contribution of a particular parameter to the total variance, including interactions with other parameters. It can be expressed by

$$TS(i) = 1 - S_{\neg i}\,, \tag{13}$$

where $S_{\neg i}$ is the sum of all $S_{i_1,\ldots,i_l}$ that does not involve parameter i.

These total sensitivity indices can be approximated using sampling based methods in order to evaluate the integrals involved in (9). Alternatively, [4] suggests to use cut-HDMR and cubature rules in the following manner:

1. Compute the cut-HDMR expansion on cubature nodes for the input distributions.
2. Derive the approximated ANOVA-HDMR expansion from the cut-HDMR.
3. Compute the Total Sensitivity Indices from the ANOVA-HDMR.

This approach gives the freedom of selecting the level of accuracy for the HDMR expansion depending on the level of interaction between parameters. The truncation order L of the ANOVA-HDMR can be selected and the accuracy of such expansion can be assessed using the concept of "effective dimension" of the system: for $q \le 1$, the effective dimension of the integrand f is an integer L such that

$$\sum_{0<|t|\le L} D_t \ge qD\,, \tag{14}$$

where t is a multi-index $i_1,\ldots,i_l$ and $|t|$ is the cardinality of such multi-index. The parameter q is chosen based on a compromise between accuracy and computational cost.

4. SENSITIVITY ANALYSIS ON RAILWAY VEHICLE DYNAMICS

The study of uncertainty propagation and sensitivity analysis through dynamical systems is a computationally expensive task. In this analysis we adopt a collocation approach, where we study the behaviors of ensembles of realizations. From the algorithmic point of view, the quality of a method is measured in the number of realizations needed in order to infer the same statistics. Each realization is the result of an Initial Value Problem (IVP) computed using the program DYnamics Train SImulation (DYTSI) developed in [5], where the model presented in section 2 has been set up and the IVP has been solved using the Explicit Runge–Kutta–Fehlberg method ERKF34 [15]. An explicit solver has been used in light of the analysis performed in [16], where it was found that the hunting motion could be missed by implicit solvers, used with relaxed tolerances, due to numerical damping. In particular implicit solvers are frequently used for stiff problems, like the one treated here, because their step-size is bounded

by accuracy constraints instead of stability. However, the detection of hunting motion requires the selection of strict tolerances, reducing the allowable step-sizes and making the implicit methods more expensive than the explicit ones. Since the collocation approach for UQ involves the computation of completely independent realizations, this allows for a straightforward parallelization of the computations on clusters. Thus, 25 nodes of the DTU cluster have been used to speed up the following analysis.

The first step in the analysis of a stochastic system is the characterization of the probability distribution of the QoI. Since the complete model has co-dimension 24, a traditional sampling method, among the ones presented in section 3.1, is the most suited for the task of approximating the integrals in eq. (4). Fig. 3 shows the histogram of the computed critical speeds with respect to the uncertainty in the suspension components. In order to speed up the convergence, we used 200 samples generated with the Latin Hyper Cube method [17]. Kernel smoothing [10] has been used to estimate the density function according to this histogram. The estimated mean and variance are $\overline{\mu}_v = 51.83 m/s$ and $\overline{\sigma}_v = 4.07 m^2/s^2$.

Suspension	One-at-a-time	ANOVA		ANOVA - Refined	
	$\overline{\sigma}_v$	$\overline{\sigma}_v$	Tot. Sensitivity	$\overline{\sigma}_v$	Tot. Sensitivity
PSLL_LEFT_K1	0.00	0.03	0.01		
PSLL_LEFT_K2	0.06	0.18	0.06	0.18	0.09
PSLL_LEFT_K3	0.02	0.13	0.04	0.14	0.07
PSLL_RIGHT_K1	0.00	0.05	0.02		
PSLL_RIGHT_K2	0.06	0.17	0.06	0.22	0.11
PSLL_RIGHT_K3	0.03	0.17	0.06	0.10	0.05
PSLT_LEFT_K1	0.00	0.02	0.01		
PSLT_LEFT_K2	0.54	1.71	0.56	1.29	0.63
PSLT_LEFT_K3	0.14	0.20	0.07	0.11	0.05
PSLT_RIGHT_K1	0.00	0.05	0.02		
PSLT_RIGHT_K2	0.55	1.73	0.56	1.22	0.59
PSLT_RIGHT_K3	0.03	0.13	0.04	0.17	0.08
SSL_LEFT_K4	0.00	0.01	0.00		
SSL_LEFT_K5	0.00	0.01	0.00		
SSL_LEFT_K6	0.00	0.02	0.01		
SSL_LEFT_D1	0.00	0.02	0.01		
SSL_LEFT_D2	0.02	0.04	0.01		
SSL_LEFT_D6	0.00	0.02	0.01		
SSL_RIGHT_K4	0.00	0.01	0.00		
SSL_RIGHT_K5	0.00	0.00	0.00		
SSL_RIGHT_K6	0.00	0.02	0.01		
SSL_RIGHT_D1	0.00	0.03	0.01		
SSL_RIGHT_D2	0.00	0.04	0.01		
SSL_RIGHT_D3	0.00	0.02	0.01		

Table 1: Variances and Total Sensitivity Indices obtained using the One-at-a-time analysis, the ANOVA expansion of the complete model and the more accurate ANOVA expansion of the reduced model. The naming convention used for the suspensions works as follows. PSL and SSL stand for primary and secondary suspension of the leading bogie respectively. The following L and T in the primary suspension stand for leading and trailing wheel sets. The last part of the nomenclature refers to the particular suspension components as shown in Fig. 1.

4.1 One-at-a-time analysis

When each suspension component is considered independently from the others, the estimation problem in (4) is reduced to the calculation of a 1-dimensional integral. This task can be readily achieved by quadrature rules that have proven to be computationally more efficient on problems of this dimensionality than sampling methods [3]. Fourth order quadrature rules have been used to approximate the variances due to the single components. The convergence of this method enables a check of accuracy through the decay of the expansion coefficients of the target function [3].

The second column in Table 1 lists the results of such analysis. The amount of variance described by this analysis is given by the sum of all the variances: $\overline{\sigma}_{\mathrm{OAT}} = 1.47 m^2/s^2$. This quantity is far from representing the total variance of the stochastic system, suggesting that interactions between suspension components are important.

Anyway the method is useful to make a first guess about which components are the most important: the critical speed of the railway vehicle model analyzed in this work shows a strong sensitivity related to the longitudinal

springs (K2) in the trailing wheel set.

4.2 Total Sensitivity Analysis

The technique outlined in section 3.3 can fulfill three important tasks: taking into account parameter interactions, performing the analysis with a limited number of realizations and enabling an error control in the approximation. In a first stage we consider the full stochastic model and we construct a cut-HDMR expansion which takes into account 2nd order interactions and describes the target function through 2nd order polynomials, computing the realizations for up to 2-dimensional cubature rules. The ANOVA-HDMR expansion of the cut-HDMR expansion can be quickly computed, due to the low dimensionality of the single terms in (10). At this point, the $D_{i_1,\ldots,i_l}$ values in (9) can be obtained and the "effective dimensionality" of the target function, given by (14) for $q = 0.95$, is found to be $L = 2$. This confirms that the 1st and 2nd order interactions are sufficient to describe most of the variance. The third and fourth column of Table 1 list the total variances induced by each parameter, including interactions with other parameters, and the Sobol' total sensitivity indices.
Once the first approximation of the sensitivities is obtained, the parameters with the lowest sensitivity indices can be fixed to their nominal values and we can perform a more accurate analysis of the remaining stochastic system. Longitudinal and vertical springs (K2 and K3) in the primary suspensions have shown to be very influential for the critical speed of the analyzed model, thus a new cut-HDMR expansion, with 2nd order interactions and 4th order polynomial approximation is constructed. The resulting total variances and total sensitivity indices are listed in the fifth and sixth column of Table 1. A visual representation of the sensitivity indices is shown in the pie chart in Fig. 4.
The results obtained by the one-at-a-time analysis are confirmed here by the total sensitivity analysis, but we stress that the latter provide a higher reliability because they describe a bigger part of the total variance of the complete stochastic system.

4.3 Discussion of the obtained results

Even if the results obtained are formally correct, the interpretation of such results can raise some questions. A railway engineer might wonder why the yaw dampers D2 are not listed among the most important by the sensitivity analysis. The yaw dampers in the secondary suspension are known to provide stability to the vehicle ride, helping to increase its critical speed. This result is true also with the vehicle model considered here, in fact low values of D2 cause a drastic worsening of the ride stability. However, the total sensitivity indices embed the probability distributions of the uncertain parameters in the global sensitivity analysis: the impact of a component is weighted according to these distributions. Thus we say that the yaw damper has little influence on the riding stability with respect to the distributions chosen. A change in the distributions can dramatically change these results, thus *particular care should be taken with the quantification of the source of uncertainty.*
Finally, observe that, even if they are not as important as the primary suspension components, the yaw dampers seem to be the most important components among the secondary suspensions.

4.4 Remarks on sensitivity analysis on non-linear dynamics

Uncertainty quantification and sensitivity analysis require a rigorous preliminary formulation of the stochastic system, its sources of uncertainty and the Quantities of Interest. We already mentioned in section 2.2 that in this work the characterization of the sources of uncertainty was bypassed by assuming Gaussian distributions for all the parameters, without loss of generality for the methods presented. The selection of the QoI, however, merits some more discussion. In section 2.1 the continuation method used to estimate the critical speed was presented and the threshold used to determine the end of the hunting motion was chosen in a conservative way, as it is shown in Fig. 2b. However, the value of the computed critical speed will depend also on the deceleration chosen for the continuation method, i.e. the computed critical speed will be exact in the limit when the deceleration goes to zero. Of course, the exact computation of the critical speed is not computationally feasible. With the limited computational resources available, we then chose a fixed deceleration coefficient for the continuation method, and thus we introduced numerical uncertainty in the computations. Therefore, the variance expressed from the analysis is given both by the variance due to the stochastic system and the variance introduced by the computation of the QoI. This is, however, a conservative consequence, meaning that a decision taken on the basis of the computed results is at least as safe as a decision taken using the "exact results".

4. CONCLUSIONS

Sensitivity analysis is of critical importance on a wide range of engineering applications. The traditional approach of local sensitivity analysis is useful in order to characterize the behavior of a dynamical system in the vicinity of the nominal values of its parameters, but it fails in describing wider ranges of variations, e.g., caused by long-term

wear. The global sensitivity analysis aims at representing these bigger variations and at the same time it embeds the probability distributions of the parameters in the analysis. This enables the engineer to take decisions based on the risk of a certain event to happen.
Wrongly approached, global sensitivity analysis can turn to be a computationally expensive or even prohibitive task. In this work a collection of techniques are used in order to accelerate such analysis for a high-co-dimensional problem. Each of the techniques used allows for a control of the accuracy, e.g., in terms of convergence rate for the cubature rules in section 3.1 and the “effective dimension” in section 3.3. This makes the framework flexible and easy to be adapted to problems with more diversified distributions and target functions.
The analysis performed on the half wagon equipped with a Cooperrider bogie shows a high importance of the longitudinal primary suspensions, and this reflects the connection between hunting and yaw motion. Furthermore, the importance of the yaw damper in the secondary suspensions is confirmed, even if its influence is little compared to the primary suspensions.
It is important to notice that the same settings for global sensitivity analysis can be used for the investigation of different Quantities of Interests, such as wear in curved tracks, angle of attack etc., once they have been properly defined. Furthermore, the “non-intrusive” approach taken allows the engineer to use closed software for the computations. The machinery for sensitivity analysis needs only to be wrapped around it, without additional implementation efforts.

References

[1] C. Funfschilling, G. Perrin and S. Kraft, "Propagation of variability in railway dynamic simulations: application to virtual homologation," *Vehicle System Dynamics*, vol. 50, no. sup1, pp. 245-261, 2012.

[2] L. Mazzola and S. Bruni, "Effect of Suspension Parameter Uncertainty on the Dynamic Behaviour of Railway Vehicles," *Applied Mechanics and Materials* , vol. 104, pp. 177-185, 2011.

[3] D. Bigoni, A. P. Engsig-Karup and H. True, "Comparison of Classical and Modern Uncertainty Quantification Methods for the Calculation of Critical Speeds in Railway Vehicle Dynamics," in *13th mini conference on vehicle system dynamics, identification and anomalies*, Budapest, 2012.

[4] Z. Gao and J. S. Hesthaven, "Efficient solution of ordinary differential equations with high-dimensional parametrized uncertainty," *Communications in Computational Physics*, vol. 10, no. 2, pp. 253-286, 2011.

[5] D. Bigoni, "Curving Dynamics in High Speed Trains," Technical University of Denmark, DTU Informatics, Kgs. Lyngby, Denmark, 2011, MSc Thesis, http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6105.

[6] W. Kik and D. Moelle, "ACRadSchiene - To create or Approximate Wheel/Rail profiles - Tutorial".

[7] J. J. Kalker, "Wheel-rail rolling contact theory," *Wear*, vol. 144, no. 1-2, pp. 243-261, 1991.

[8] Z. Y. Shen, J. K. Hedrick and J. A. Elkins, "A comparison of alternative creep-force models for rail vehicle dynamic analysis," in *8th IASVD Symposium*, 1984.

[9] H. True, "Multiple attractors and critical parameters and how to find them numerically: the right, the wrong and the gambling way," *Vehicle System Dynamics*, pp. 1-17, 2012.

[10] T. Hastie, R. Tibishirani and J. Firedman, "Kernel Smoothing Methods," in *The elements of statistical learning*, 10th ed., 2013, pp. 191-218.

[11] W. Gautschi, Orthogonal Polynomials: Computation and Approximation, Oxford: Oxford University Press, 2004.

[12] K. Petras, "Smolyak cubature of given polynomial degree with few nodes for increasing dimension," *Numerische Mathematik* , vol. 93, no. 4, pp. 729-753 , 2003.

[13] H. Rabitz and F. A. Ömer, "Managing the Tyranny of Parameters in Mathematical Modelling of Physical Systems," in *Sensitivity Analysis*, Chichester, West Sussex: John Wiley & Sons Ltd., 2000, pp. 199-223.

[14] K. Chan, S. Tarantola, A. Saltelli and I. M. Sobol', "Variance-Based Methods," in *Sensitivity Analysis*, Chichester, West Sussex: John Wiley & Sons Ltd., 2000, pp. 168-197.

[15] E. Hairer, S. P. Nørsett and G. Wanner, Solving Ordinary Differential Equations I: Nonstiff problems, second revision ed., Berlin, Wien, New York: Springer Series in Computational Mathematics, Springer-Verlag, 1991.

[16] H. True, A. P. Engsig-Karup and D. Bigoni, "On the Numerical and Computational Aspects of Non-Smoothnesses that occur in Railway Vehicle Dynamics," *Mathematics and Computers in Simulation*, 2012.

[17] M. D. McKay, R. J. Beckman and W. J. Conover, "A comparison of three methods of selecting values of input variables in the analysis of output from a computer code.," *Technometrics*, vol. 21, pp. 239-245, 1979.

Proceedings of the ASME 2013 Rail Transportation Division Fall Technical Conference
RTDF2013
October 15-17, 2013, Altoona, Pennsylvania, USA

RTDF2013-4713

MODERN UNCERTAINTY QUANTIFICATION METHODS IN RAILROAD VEHICLE DYNAMICS

D. Bigoni, A.P. Engsig-Karup, and H. True
Department of Applied Mathematics and Computer Science
The Technical University of Denmark
DK-2800 Kgs. Lyngby, Denmark

ABSTRACT

This paper describes the results of the application of Uncertainty Quantification methods to a simple railroad vehicle dynamical example. Uncertainty Quantification methods take the probability distribution of the system parameters that stems from the parameter tolerances into account in the result. In this paper the methods are applied to a low-dimensional vehicle dynamical model composed by a two-axle truck that is connected to a car body by a lateral spring, a lateral damper and a torsional spring, all with linear characteristics.

Their characteristics are not deterministically defined, but they are defined by probability distributions. The model - but with deterministically defined parameters - was studied in [1] and [2], and this article will focus on the calculation of the critical speed of the model, when the distribution of the parameters is taken into account.

Results of the application of the traditional Monte Carlo sampling method will be compared with the results of the application of advanced Uncertainty Quantification methods [3]. The computational performance and fast convergence that result from the application of advanced Uncertainty Quantification methods is highlighted. Generalized Polynomial Chaos will be presented in the Collocation form with emphasis on the pros and cons of each of those approaches.

NOMENCLATURE

m, I	mass and inertia of the bogie
D_2, k_4, k_6	suspension parameters
$b, k_0, x_f, \alpha,$ β, δ, κ	nonlinear spring constants used to approximate the flange forces
ϕ, ψ	constants determined by the sizes of the semi axes of the contact ellipse
r_0	nominal rolling radius
λ	conicity

INTRODUCTION

In engineering, deterministic models have been extensively exploited to describe dynamical systems and their behaviors. These have proven to be useful in the design phase of the engineering products, but they always fall short in providing indications of the reliability of certain designs over others. The results obtained by one deterministic experiment describe, in practice, a very rare case that likely will never happen. However, engineers are confident that this experiment will explain most of the experiments in the vicinity of it, i.e. for small variation of parameters. Unfortunately, this assumption may lead to erroneous conclusions, in particular for realistic nonlinear dynamical systems, where small perturbations can cause dramatic changes in the dynamics. It is thus critical to find a measure for the level of knowledge of a dynamical system, in order to be able to make a reasonable risk analysis and design optimization.

Risk analysis in the railroad industry is critical for as well the increase of the safety as for targeting investments. Railroad vehicle dynamics is difficult to study even in the deterministic case, where strong nonlinearities appear in the system. A lot of phenomena develop in such dynamical systems, and the interest of the study could be focused on different parameters, such as the ride comfort or the wear of the components. This work will instead focus on ride safety when high speeds are reached and the hunting motion develops. The hunting motion is a well known phenomenon characterized by periodic as well as aperiodic lateral oscillations, due to the wheel-rail contact forces, that can appear at different speeds depending on the vehicle design. This motion can be explained and studied with notions from nonlinear dynamics [4], combined with suitable numerical methods for non-smooth dynamical systems [5]. It is well known that the behavior of the hunting motion is parameter dependent, thus good vehicle designs can increase the critical speed. This also means that suspension components

Copyright © 2013 by ASME

need to be carefully manufactured in order to really match the demands of the customer. However, no manufactured component will ever match the simulated ones. Thus epistemic uncertainties, for which we have no evidence, and aleatoric uncertainties, for which we have a statistical description, appear in the system as a level of knowledge of the real parameters [6].

Uncertainty Quantification (UQ) tries to address the question: "assuming my partial knowledge of the design parameters, how reliable are my results?". This work will focus on the sensitivity of the critical speed of a railroad vehicle model to the suspension parameters.

THE VEHICLE MODEL

This work will investigate the dynamics of the well known simple Cooperrider truck model [2] shown in Fig. 1. The model is composed by two conical wheel sets rigidly connected to a truck frame, that is in turn connected to a fixed car body by linear suspensions: a couple of lateral springs and dampers and one torsional spring.

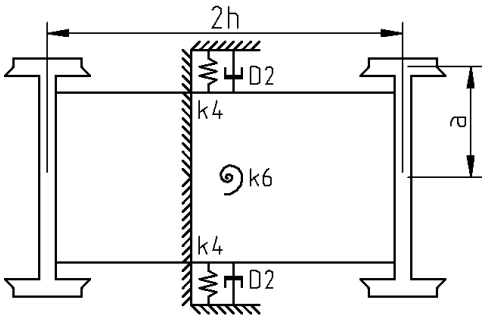


Fig. 1: Top view of the Cooperrider truck model.

The following equations govern this dynamical system [2]:

$$\begin{aligned} m\ddot{q}_1 = &-2D_2\dot{q}_1 - 2k_4 q_1 \\ &- 2\left[F_x\left(\xi_{x_1},\xi_{y_1}\right) + F_x\left(\xi_{x_2},\xi_{y_2}\right)\right] \\ &- F_T(q_1 + haq_2) - F_T(q_1 - haq_2), \\ I\ddot{q}_2 = &-k_6 q_2 - 2ha\left[F_x\left(\xi_{x_1},\xi_{y_1}\right) - F_x\left(\xi_{x_2},\xi_{y_2}\right)\right] \\ &- 2a\left[F_y\left(\xi_{x_1},\xi_{y_1}\right) + F_y\left(\xi_{x_2},\xi_{y_2}\right)\right] \\ &- ha[F_T(q_1 + haq_2) \\ &- F_T(q_1 - haq_2)], \end{aligned} \tag{1}$$

where D_2, k_4 and k_6 are the damping coefficient and the stiffness coefficients respectively, F_x and F_y are the lateral and longitudinal creep forces and F_T is the flange force.

The ideally stiff truck runs on a perfect straight track where the constant wheel-rail adhesion coefficient enters the system through the lateral and longitudinal creep-forces:

$$F_x\left(\xi_x,\xi_y\right) = \frac{\xi_x}{\phi}\frac{F_R(\xi_x,\xi_y)}{\xi_R(\xi_x,\xi_y)} \quad , \quad F_y\left(\xi_x,\xi_y\right) = \frac{\xi_y}{\psi}\frac{F_R(\xi_x,\xi_y)}{\xi_R(\xi_x,\xi_y)} \ ,$$

$$\xi_R\left(\xi_x,\xi_y\right) = \sqrt{\frac{\xi_x^2}{\phi^2} + \frac{\xi_y^2}{\psi^2}} \ ,$$

$$\frac{F_R\left(\xi_x,\xi_y\right)}{\mu N} = \begin{cases} u(\xi_R) - \frac{1}{3}u^2(\xi_R) + \frac{1}{27}u^3(\xi_R) & \text{for } u(\xi_R) < 3 \\ 1 & \text{for } u(\xi_R) \geq 3 \end{cases} ,$$

$$u(\xi_R) = \frac{G\pi ab}{\mu N}\xi_R \ ,$$

where ϕ and ψ are real numbers that are determined by the size of the semi axes of the contact ellipse, which are constant in our problem [7]. The creepages are given by:

$$\xi_{x_1} = \frac{\dot{q}_1}{v} + ha\frac{\dot{q}_2}{v} - q_2 \ , \qquad \xi_{y_1} = a\frac{\dot{q}_2}{v} + \frac{\lambda}{r_0}(q_1 + haq_2) \ ,$$

$$\xi_{x_2} = \frac{\dot{q}_1}{v} - ha\frac{\dot{q}_2}{v} - q_2 \ , \qquad \xi_{y_2} = a\frac{\dot{q}_2}{v} + \frac{\lambda}{r_0}(q_1 - haq_2) \ .$$

The flange forces are approximated by a very stiff non-linear spring with a dead band:

$$F_T(x) = \begin{cases} \exp\left(-\alpha/\left(x - x_f\right)\right) - \beta x - \kappa \ , & 0 \leq x < b \\ k_0 \cdot (x - \delta) \ , & b \leq x \\ -F_T(-x) \ , & x < 0 \end{cases} ,$$

The parameters used for the analysis are listed in the following:

$m = 4963\ kg$	$h = 1.5\ m$
$I = 8135\ kg \cdot m^2$	$D_2 = 29200\ N \cdot s/m$
$k_0 = 14.60 \cdot 10^6\ N/m$	$k_4 = 0.1823 \cdot 10^6\ N/m$
$k_6 = 2.710 \cdot 10^6\ N/m$	$\lambda = 0.05$
$r_0 = 0.4572\ m$	$b = 0.910685 \cdot 10^{-2}\ m$
$\phi = 0.60252$	$\psi = 0.54219$
$G\pi ab = 6.563 \cdot 10^6\ N$	$\mu N = 10^4\ N$
$\delta = 0.0091\ m$	$\alpha = 0{,}1474128791 \cdot 10^{-3}$
$\beta = 1{,}016261260$	$\kappa = 1{,}793756792$
$x_f = 0.9138788366 \cdot 10^{-2}$	$a = 0.7163\ m$

Non linear dynamics of the deterministic model

The dynamics of the deterministic model at high speed has been investigated in [2]. The existence of a subcritical Hopf-bifurcation has been detected at $v_L = 66.61$ m/s. Fig. 2 shows the bifurcation diagram of the deterministic system. The Hopf bifurcation point is obtained by observation of the stability of the trivial solution using the eigenvalues of the Jacobian of the system. The nonlinear critical speed, the fold bifurcation, characteristic in subcritical Hopf-bifurcations, is found at $v_{NL} = 62.02$m/s using a ramping method, where the speed is quasi-statically decreased, according to

$$\dot{v} = \begin{cases} 0 \ , & \text{if } t < t_{st} \vee \|\vec{q}\|_2 < \epsilon_{min} \\ -\Delta \ , & \text{otherwise} \end{cases} . \tag{2}$$

The stochastic model

Let us now consider suspensions that are provided by the manufacturer with a certain level of working accuracy. Due to the lack of real data regarding the probability distributions of

Copyright © 2013 by ASME

such working accuracies, this initial study will consider Gaussian distributions to describe them:

$$\begin{aligned} k_6 &\sim \mathcal{N}\left(\mu_{k_6}, \sigma_{k_6}^2\right) , \quad (\text{std.} \sim 5\%) \\ k_4 &\sim \mathcal{N}\left(\mu_{k_4}, \sigma_{k_4}^2\right) , \quad (\text{std.} \sim 7\%) \\ D_2 &\sim \mathcal{N}\left(\mu_{D_2}, \sigma_{D_2}^2\right) . \quad (\text{std.} \sim 7\%) \end{aligned} \tag{3}$$

where the symmetry of the model is taken into consideration in the standard deviation of the parameters k_4 and D_2 that both represent two elements. The applicability and efficiency of the methods presented in the next section will not be affected by the particular choice of distribution.

Now the deterministic model is turned into a stochastic model, where the single solution represents a particular realization and probabilistic moments can be used to describe the statistics of the stochastic solution.

A straightforward way of computing the moments of the solution is to approximate the integrals as:

$$\begin{aligned} \boldsymbol{\mu_q}(t) &\approx \bar{\boldsymbol{\mu}}_{\boldsymbol{q}}(t) = \frac{1}{M}\sum_{j=1}^{M} \boldsymbol{q}\left(t, \boldsymbol{Z}^{(j)}\right) , \\ \boldsymbol{\sigma}_{\boldsymbol{q}}^2(t) &\approx \bar{\boldsymbol{\sigma}}_{\boldsymbol{q}}^2(t) = \frac{1}{M-1}\sum_{j=1}^{M} \left(\boldsymbol{q}\left(t, \boldsymbol{Z}^{(j)}\right) - \bar{\mu}_{\boldsymbol{Z}}(t)\right)^2 , \end{aligned} \tag{5}$$

where $\left\{\boldsymbol{Z}^{(j)}\right\}_{j=1}^{M}$ are realizations sampled randomly from the probability distribution of $\boldsymbol{Z}$. This is the Monte-Carlo (MC) method and it has a probabilistic error of $\mathcal{O}\left(1/\sqrt{M}\right)$.

Even though the MC methods are really robust and versatile, such a slow convergence rate is problematic, when the solution of a single realization of the system is computationally expensive. Alternative sampling methods are the Quasi Monte-Carlo methods (QMC). These can provide

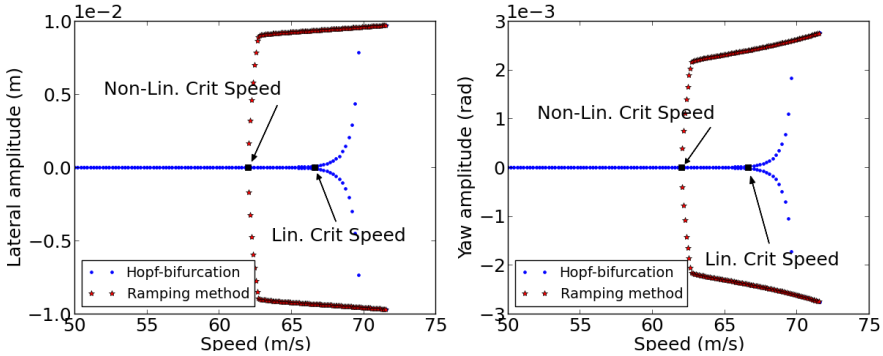


Fig. 2: Non-linear dynamics of the deterministic system. The subcritical Hopf-bifurcation is highlighted and the critical speed is determined exactly at $v_L = 66.61\, m/s$. The ramping method is then used in order to detect the non-linear critical speed at $v_{NL} = 62.02\, m/s$.

UNCERTAINTY QUANTIFICATION

The stochastic solution of the system is now represented by $\boldsymbol{q}(t, \boldsymbol{Z})$, where $\boldsymbol{Z}$ is a vector of random variables distributed according to (3). The solution is a function that spans over a three dimensional random parameter space. The dimension of the parameter space is called *the co-dimension* of the dynamical problem. In this work the focus will be restricted to the first two moments of this solution, namely the mean $\boldsymbol{E}[\boldsymbol{q}(t, \boldsymbol{Z})]$ and variance $\boldsymbol{V}[\boldsymbol{q}(t, \boldsymbol{Z})]$, but the following derivations can be used similarly for higher moments too. Mean and variance are defined as

$$\begin{aligned} \boldsymbol{\mu_q}(t) &= \boldsymbol{E}[\boldsymbol{q}(t, \boldsymbol{Z})]_{\rho_{\boldsymbol{Z}}} = \iiint \boldsymbol{q}(t, \boldsymbol{z}) \rho_{\boldsymbol{Z}}(\boldsymbol{z}) d\boldsymbol{z} , \\ \boldsymbol{\sigma}_{\boldsymbol{q}}^2(t) &= \boldsymbol{V}[\boldsymbol{q}(t, \boldsymbol{Z})]_{\rho_{\boldsymbol{Z}}} = \iiint \left(\boldsymbol{q}(t, \boldsymbol{z}) - \mu_{\boldsymbol{q}}(t)\right)^2 \rho_{\boldsymbol{Z}}(\boldsymbol{z}) d\boldsymbol{z} \end{aligned} \tag{4}$$

where $\rho_{\boldsymbol{Z}}(\boldsymbol{z})$ is the probability density function of the random vector $\boldsymbol{Z}$ and the integrals are computed over its domain.

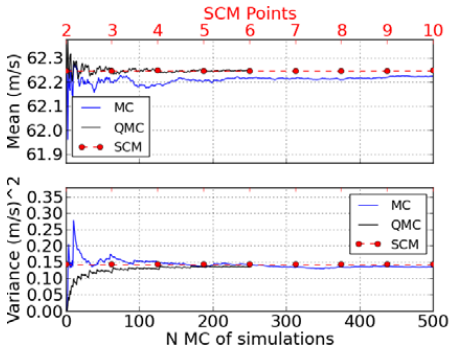
convergence rates of $\mathcal{O}((\log M)^d / M)$, where d is the co-dimension of the problem. They use low discrepancy sequences in order to uniformly cover the sampling domain. Without presumption of completeness, in this work only the Sobol sequence will be considered as a measure of comparison with respect to other advanced UQ methods. QMC methods are known to work better than MC methods when the integrand is sufficiently smooth, whereas they can completely fail on an integrand of unbounded variation [8]. Furthermore, randomized versions of the QMC method are available in order to improve the variance estimation of the method.

Stochastic collocation method (SCM)

Collocation methods require the residual of the governing equations to be zero at the collocation points $\left\{\boldsymbol{Z}^{(j)}\right\}_{\boldsymbol{j=1}}^{\boldsymbol{Q}}$, i.e.

$$\begin{cases} \partial_t \boldsymbol{q}\left(t, \boldsymbol{Z}^{(j)}\right) = \mathcal{L}\left(\boldsymbol{q}\left(t, \boldsymbol{Z}^{(j)}\right)\right), & (0, T] \\ \boldsymbol{q}(0) = \boldsymbol{q}_0, & t = 0 \ . \end{cases} \tag{6}$$

Copyright © 2013 by ASME



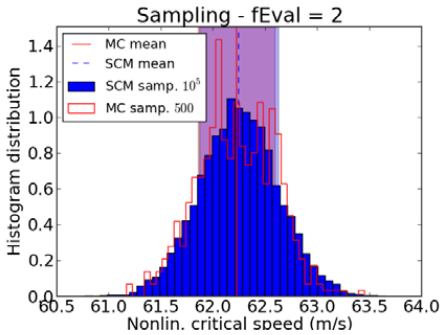


Fig. 3: SCM on the model with 1D uncertainty on parameter k_4 compared with MC and QMC. Left, estimation of mean and variance of the nonlinear critical speed. Right, histograms of NL critical speeds obtained using 500 MC simulations of model (1)-(2) and 10^5 realizations using the approximated stochastic solution (7) with only 2 function evaluations. The standard deviation is shown as a shaded confidence interval, blue for SCM and red for MC. The two confidence intervals are overlapping almost exactly.

Then an approximation $\boldsymbol{w}(t,\boldsymbol{Z})$ of $\boldsymbol{q}(t,\boldsymbol{Z})$ is found as an expansion in a set of Hermite polynomials, which are suitable for approximations of the Gauss distribution functions:

$$\boldsymbol{w}_N(t,\boldsymbol{Z}) = \sum_{|k|\leq N} \widehat{\boldsymbol{w}}_k(t)\,\mathcal{H}_k(\boldsymbol{Z})\ ,$$

$$\widehat{\boldsymbol{q}}_k = \frac{1}{\gamma_k}\iiint \boldsymbol{q}(t,\boldsymbol{z})\,\mathcal{H}_k(\boldsymbol{z})\rho_{\boldsymbol{Z}}(\boldsymbol{z})d\boldsymbol{z} \approx \widehat{\boldsymbol{w}}_k = \frac{1}{\gamma_k}\sum_{j=1}^{Q} \boldsymbol{q}\big(t,\boldsymbol{z}^{(j)}\big)\,\mathcal{H}_k\big(\boldsymbol{z}^{(j)}\big)\alpha^{(j)}\ , \tag{7}$$

where we used a cubature rule with points and weights $\{\boldsymbol{z}^{(j)},\alpha^{(j)}\}_{j=1}^{Q}$. The points $\{\boldsymbol{z}^{(j)}\}_{j=1}^{Q}$ are the set of parameter values for which deterministic solutions must be computed. Cubature rules with different accuracy levels and sparsity exist. In this work simple tensor product structured Gauss cubature rules will be used. These are the most accurate but scale with $\mathcal{O}(m^d)$, where m is the number of points in one dimension and d is the co-dimension. The fast growth of the number of collocation points with the dimensionality goes under the name of "the curse of dimensionality" and can be addressed using more efficient cubature rules such as Smolyak sparse grids [9].

UNCERTAINTY QUANTIFICATION IN RAILROAD VEHICLE DYNAMICS

Uncertainty quantification is recently gaining much attention from many engineering fields and in vehicle dynamics there are already some contributions on the topic. In [10] a railroad vehicle dynamic problem with uncertainty on the suspension parameters was investigated using MC method coupled with techniques from Design of Experiments.

Here SCM will be applied to the simple Cooperrider truck [2] in order to study its behavior with uncertainties, and the results will be compared to the ones obtained by the MC and QMC methods.

These methods belong to the class of non-intrusive methods for Uncertainty Quantification. This means that they only require a deterministic method to compute the quantity of interest (QoI) for different parameters. In this work this is the ramping method to detect the critical speed.

The focus of this work is on the determination of the nonlinear critical speed with uncertainties, so the investigation of the stochastic dynamics with respect to time will be disregarded here. Fig. 3 shows the SCM method applied to the model with 1D uncertainty on parameter k_4, for the determination of the first two moments of the nonlinear critical speed. The estimation done by the SCM is already satisfactory at low order and little is gained by increasing it. This means that the few first terms of the expansion (7) are sufficient in approximating the nonlinear critical speed distribution.

Fig. 4 shows the SCM method applied to the same problem with 1D uncertainty on the torsional spring stiffness k_6. Again the first few terms in expansion (7) are sufficient in order to give a good approximation of the nonlinear critical speed distribution. It is worth noting that the torsional spring stiffness k_6 has an higher influence on the critical speed than k_4.

Fig. 5 shows the SCM method on the problem with uncertainty on parameters k_6, k_4 and D_2. Again, a low-order SCM approximation is sufficient to get the most accurate solution.

In the figures 3-5, left, we have compared the convergence of the SCM method with that of the MC method. Therefore the number of evaluations was prescribed. It is also of interest to compare the computation time of the methods expressed by the CPU time. For the comparison we used the calculated mean values of the critical speed as the basis for the comparison. For the SCM method the iteration process was ended when the second decimal remained constant. The mean values in the MC and QMC methods change however a good deal as shown in the figures 3-5, left. Therefore, for the comparison a window with 20 iterative values, which is glided over the number of iterations was used. When the second decimal of the average of

Copyright © 2013 by ASME

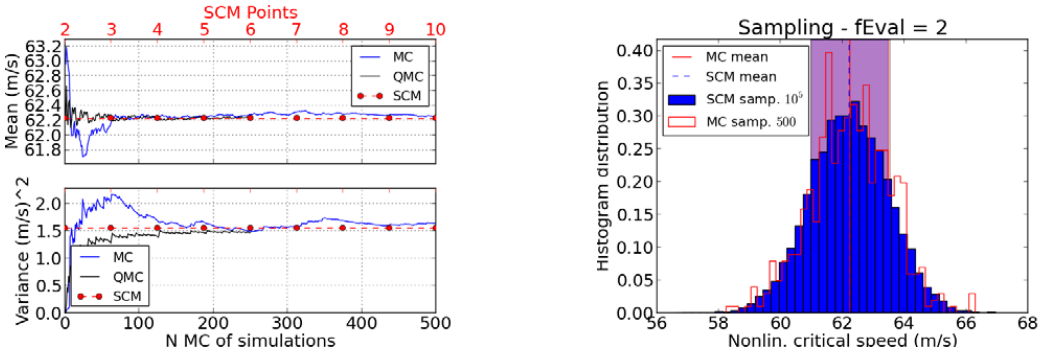


Fig. 4: SCM on 1D uncertainty on parameter k_6 compared with MC and QMC. Left, estimation of mean and variance of the non-linear critical speed. Right, histograms of NL critical speeds obtained using 500 MC simulations of model (1)-(2) and 10^5 realizations using the approximated stochastic solution (7) with only 2 function evaluations. The standard deviation is shown as a shaded confidence interval, blue for SCM and red for MC. The two confidence intervals are overlapping almost exactly.

the iterated values in the window remained constant, when the widow was pushed one more step, then the iterations were stopped, and the CPU time was stored.

Table 1 shows the final results obtained with the chosen accuracy, using the three methods, Monte-Carlo (MC), Quasi-Monte-Carlo (QMC) and Stochastic Collocation (SCM). We can observe that the variances in the multi-dimensional cases are almost equal to the sum of the single-dimensional cases. This means that the effect of the nonlinear interactions between the three elements of the suspension is small with the variances chosen in this problem.

CONCLUSIONS

Manufacturing tolerances have been introduced into the dynamical investigations of vehicles. A new method, the Stochastic Collocation Method (SCM) is applied as a tool for "Uncertainty Quantification", and the accuracy and computational effort is compared with that of Monte-Carlo (MC) and Quasi-Monte-Carlo (QMC) methods. The "Uncertainty Quantification" methods are applied to the estimate of the calculated critical speed of a railroad vehicle model. The critical speed is delivered as a mean value with variance. The results show that under the condition of the same accuracy the convergence rate of the SCM outperforms the rates of as well the MC as the QMC methods. Table 1 shows that the CPU time and thus the computational effort by application of the SCM is much smaller than the computational effort by application of the MC or QMC methods. By all three methods the total computational effort is larger than the effort by a deterministic computation, because the same dynamical system must be solved repeatedly only with different parameter values. Under these conditions it is however possible to reduce the total *elapsed time* significantly by straightforward application of *parallel computing*. The dynamics of the vehicle model is calculated in the process, but the results are not shown here due to the limited space. A very simple model was chosen in order to demonstrate the superiority of SCM over the MC

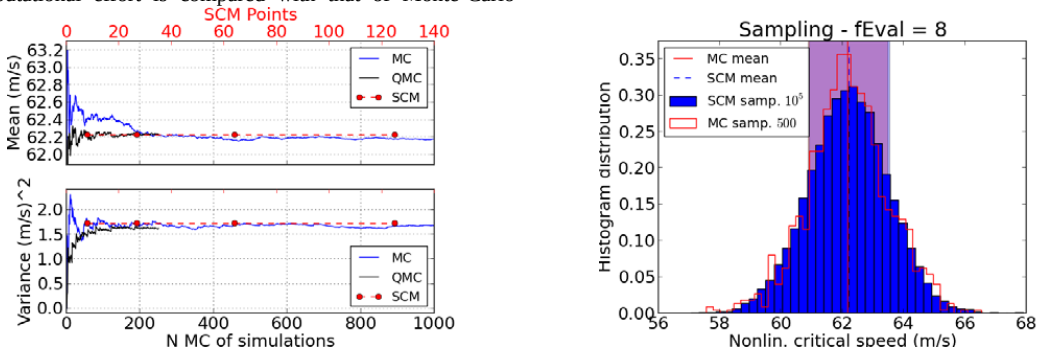


Fig. 5: SCM on 3D uncertainty compared with MC and QMC. Left, estimation of the mean and variance of the non-linear critical speed. Right, histograms of nonlinear critical speeds.

Copyright © 2013 by ASME

	MC				QMC				SCM			
	μ	σ^2	**#fE**	**CPUt**	μ	σ^2	**#fE**	**CPUt**	μ	σ^2	**#fE**	**CPUt**
k_6	62,26	1,64	169	∼24h	62,24	1,47	152	∼21 h	62,23	1,55	2	∼10m
k_4	62,23	0,14	17	∼2,5h	62,25	0,14	22	∼3 h	62,25	0,14	2	∼11m
D_2	62,23	0,02	9	∼1 h	62,25	0,02	4	∼30m	62,25	0,03	2	∼11m
k_6, k_4	62,22	1,53	148	∼21 h	62,22	1,62	152	∼22 h	62,28	1,69	4	∼36m
k_6, D_2	62,18	1,72	216	∼30 h	62,24	1,50	142	∼20 h	62,28	1,57	4	∼37m
k_4, D_2	62,25	0,17	25	∼3,5h	62,25	0,16	25	∼3,5h	62,30	0,17	4	∼35m
k_6, k_4, D_2	62,18	1,68	221	∼32 h	62,23	1,63	154	∼22 h	62,23	1,72	8	∼1 h

Table 1: Estimated mean and variance of the nonlinear critical speed using MC, QMC and SCM. The three methods are compared in terms of number of function evaluations (#fE) and computation time (CPUt).

and QMC methods. By using the same distributions for the characteristics of the two lateral springs and dampers the effect of the loss of symmetry in a real vehicle was not investigated here. SCM can be 100 times faster than MC for low co-dimensional problems, but for high co-dimensional problems SCM methods suffer from the “curse of dimensionality”. The computational effort of the SCM grows very fast with the number of independent parameters. In a realistic vehicle model that number easily surpasses 20. Therefore the work continues with an investigation of the application of statistical methods that may reduce the computational effort by singling out the parameters that have the most important influence on the wanted result of the dynamical problem. Some early results are shown in [11].

REFERENCES

[1] N. Cooperrider, "The hunting behavior of conventional railway trucks," *ASME Journal of Engineering and Industry*, vol. 94, pp. 752-762, 1972.

[2] H. True and C. Kaas-Petersen, "A Bifurcation Analysis of Nonlinear Oscillations in Railway Vehicles," in *Proc. 8th IAVSD-IUTAM Symposium on Vehicle System Dynamics*, Lisse, 1984.

[3] D. Xiu, Numerical Methods for Stochastic Computations: A Spectral Method Approach, Princeton: Princeton University Press, 2010.

[4] H. True, "On the Theory of Nonlinear Dynamics and its Applications in Vehicle Systems Dynamics," *Vehicle System Dynamics* , vol. 31, pp. 393-421, 1999.

[5] H. True, A. P. Engsig-Karup and D. Bigoni, "On the Numerical and Computational Aspects of Non-Smoothnesses that occur in Railway Vehicle Dynamics," *Mathematics and Computers in Simulation*, 2012.

[6] S. F. Wojtkiewicz, M. S. Eldred, R. V. Field, A. Urbina and J. R. Red-Horse, "Uncertainty Quantification In Large Computational Engineering Models," *American Institute of Aeronautics and Astronautics* , vol. 14, 2001.

[7] P. J. Vermeulen and K. L. Johnson, "Contact of nonspherical elastic bodies transmitting tangential forces," *Journal of Applied Mathematics*, vol. 31, pp. 338-340, 1964.

[8] W. J. Morokoff and R. E. Caflisch, "Quasi-Monte Carlo Integration," *Journal of Computational Physics* , vol. 122, no. 2, pp. 218-230 , 1995.

[9] K. Petras, "Smolyak cubature of given polynomial degree with few nodes for increasing dimension," *Numerische Mathematik* , vol. 93, no. 4, pp. 729-753 , 2003.

[10] L. Mazzola and S. Bruni, "Effect of Suspension Parameter Uncertainty on the Dynamic Behaviour of Railway Vehicles," *Applied Mechanics and Materials* , vol. 104, pp. 177-185 , 2011.

[11] D. Bigoni, H. True and A. P. Engsig-Karup, "Sensitivity analysis of the critical speed in railway vehicle dynamics," in *23rd International Symposium on Dynamics of Vehicles on Roads and Tracks*, Qingdao, China, 2013.

Copyright © 2013 by ASME

Vehicle System Dynamics, 2014
http://dx.doi.org/10.1080/00423114.2014.898776

Taylor & Francis
Taylor & Francis Group

Sensitivity analysis of the critical speed in railway vehicle dynamics

D. Bigoni*, H. True and A.P. Engsig-Karup

DTU Compute, The Technical University of Denmark, 303B Matematiktorvet, DK-2800 Kgs Lyngby, Denmark

(*Received 25 October 2013; accepted 22 February 2014*)

We present an approach to global sensitivity analysis aiming at the reduction of its computational cost without compromising the results. The method is based on sampling methods, cubature rules, high-dimensional model representation and total sensitivity indices. It is applied to a half car with a two-axle Cooperrider bogie, in order to study the sensitivity of the critical speed with respect to the suspension parameters. The importance of a certain suspension component is expressed by the variance in critical speed that is ascribable to it. This proves to be useful in the identification of parameters for which the accuracy of their values is critically important. The approach has a general applicability in many engineering fields and does not require the knowledge of the particular solver of the dynamical system. This analysis can be used as part of the virtual homologation procedure and to help engineers during the design phase of complex systems.

Keywords: reliability analysis; uncertain dynamics; vehicle safety; bifurcation analysis

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

1. Introduction

The past couple of decades have seen the advent of computer simulations for the study of deterministic dynamical systems arising in any field of engineering. The reasons behind this trend are both the enhanced design capabilities during production and the possibility of understanding dangerous phenomena. However, deterministic dynamical systems fall short in the task of giving a complete picture of reality: several sources of uncertainty can be present when the system is designed and thus obtained results refer to single realisations that in a probabilistic sense have measure zero, i.e. they never happen in reality. The usefulness of these simulations is, however, proved by the achievements in computer-aided design. The studies of stochastic dynamical systems allow for a wider analysis of phenomena: deterministic systems can be extended with prior knowledge on uncertainties with which the systems are described. This enables an enhanced analysis and can be used for risk assessment subject to such uncertainties and is useful for decision-making in the design phase. In the railway industry, stochastic dynamical systems are being considered in order to include their analysis as a part of the virtual homologation procedure,[1] by means of the framework for global

*Corresponding author. Email: dabi@dtu.dk

© 2014 Taylor & Francis

parametric uncertainty analysis proposed by the OpenTURNS consortium. This framework splits the uncertainty analysis task in four steps:

(a) deterministic modelling and identification of quantities of interest (QoI) and source of uncertainties;
(b) quantification of uncertainty sources by means of probability distributions;
(c) uncertainty propagation through the system and
(d) sensitivity analysis.

Railway vehicle dynamics can include a wide range of uncertainty sources. Suspension characteristics are only known within a certain tolerance when they exit the manufacturing factory and are subject to wear over time that can be described stochastically. Other quantities that are subject to uncertainties are the mass and inertia of the bodies, e.g. we do not know exactly how the wagon will be loaded, the wheel and track geometries, which are subject to wear over time, and also external loadings like wind gusts.

In this work the QoI will be the *critical speed* of a fixed half-wagon with respect to uncertain suspension components – step (a). The deterministic and stochastic models will be presented in Section 2. Step (b) requires measurements of the input uncertainty that are not available to the authors, so the probability distribution of the suspension components will be assumed to be Gaussian, without losing the generality of application of the methods used in (c) and (d). Techniques for uncertainty quantification (UQ) will be presented in Section 3.1. They have already been applied in [2,3] to perform an analysis of uncertainty propagation – step (c). They will turn useful also in Sections 3.2 and 3.3 for the sensitivity analysis technique to be presented – step (d). This is based on total sensitivity indices (TSIs) obtained from the analysis of variance (ANOVA) expansion of the function associated with the QoI.[4] Section 4 will contain the results of such analysis.

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

2. The vehicle model

In this work, we will consider a fixed half-wagon equipped with a Cooperrider bogie,[5] running on a tangent track with wheel profile S1002 and rail UIC60. The position of the suspension components is shown in Figure 1. The original design of the Cooperrider bogie included a torsional spring among the secondary suspensions, connected vertically from the geometrical centre of the bogie to the car body, in order to counteract the yaw motion. The design used in this work substitute such spring with two yaw springs that execute an equivalent torsional resistance to the original model. Thus, the spring K6 and the yaw damper D6 are mounted in parallel in this setting. See Tables 1 and 2 for the list of parameters of the model used in this work. In [6], a framework for the simulation of the dynamics of complete wagons running on straight and curved tracks has been implemented and tested based on the Newton–Euler formulation of the dynamical system:

$$\begin{aligned} \sum_{i=1}^{n} \vec{F}_i &= m\vec{a}, \\ \sum_{i=1}^{m} M_i &= \frac{\mathrm{d}}{\mathrm{d}t}([J]\cdot\vec{\omega}) + \vec{\omega}\times([J]\cdot\vec{\omega}), \end{aligned} \tag{1}$$

where F_i and M_i are, respectively, the forces and torques acting on the bodies, m and $[J]$ are the mass and inertia of the bodies, $\vec{a}$ is the acceleration and $\vec{\omega}$ is the angular velocity of the bodies.

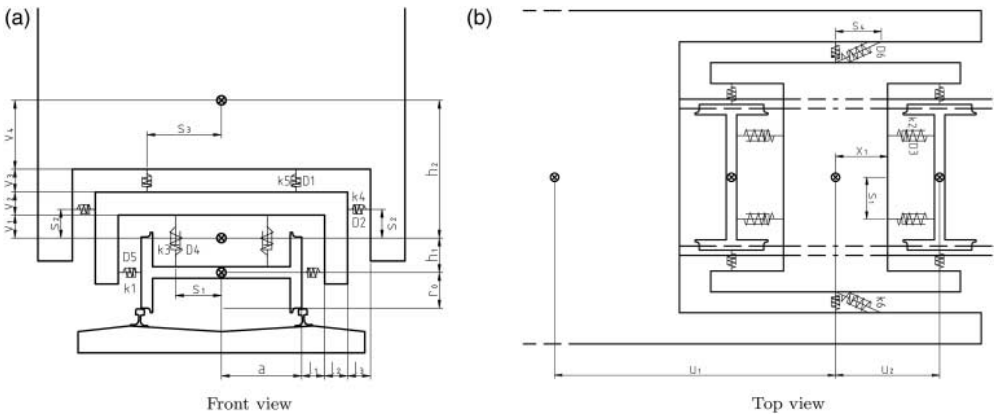


Figure 1. The half-wagon equipped with the Cooperrider bogie. (a) Front view and (b) top view.

Table 1. Dimension (see Figure 1), mass and inertia values for the components of the Cooperrider model.

Parm.	Value	Unit	Parm.	Value	Unit
r_0	0.425	(m)	a	0.75	(m)
h_1	0.0762	(m)	h_2	1.5584	(m)
l_1	0.30	(m)	l_2	0.30	(m)
l_3	0.30	(m)	x_1	0.349	(m)
v_1	0.6488	(m)	v_2	0.30	(m)
v_3	0.30	(m)	v_4	0.3096	(m)
s_1	0.62	(m)	s_2	0.6584	(m)
s_3	0.68	(m)	s_4	0.759	(m)
u_1	7.5	(m)	u_2	1.074	(m)
m_f	2918.0	(kg)	$I_{\mathrm{f}x}$	6780.0	(kg m^2)
I_fy	6780.0	(kg m^2)	$I_{\mathrm{f}z}$	6780.0	(kg m^2)
m_w	1022.0	(kg)	I_{wx}	678.0	(kg m^2)
I_wy	80.0	(kg m^2)	I_{wz}	678.0	(kg m^2)

Note: The subscript f stands for bogie frame, whereas w stands for wheel set. The nominal values of the suspension components are listed in the first column of Table 2.

In this work, the wagon will be fixed in order to alleviate the lateral oscillations during the hunting motion that would, in some cases, break the computations. The mathematical analysis and the generality of the methods proposed are not weakened by this assumption, even if the results may change for different settings.

Since we are considering a wagon running at quasi-constant speed, the longitudinal motion of the bodies has been neglected in the model. The motion of the bogie frame is then modelled using lateral, vertical and angular degrees of freedoms, with the following equations of motion:

$$\begin{aligned} m\ddot{\vec{x}} &= {}^{F}\vec{F}_{\mathrm{g}}{}^{B_\mathrm{l}} + {}^{F}\vec{F}_{\mathrm{c}}{}^{B_\mathrm{l}} + {}^{F}\vec{F}_{\mathrm{s}}{}^{\mathrm{SS_l}} + {}^{F}\vec{F}_{\mathrm{s}}{}^{\mathrm{PS_{ll}}} + {}^{F}\vec{F}_{\mathrm{s}}{}^{\mathrm{PS_{lt}}}, \\ [J]\dot{\vec{\omega}} &= {}^{B}\vec{M}_{\mathrm{g}}{}^{B_\mathrm{l}} + {}^{B}\vec{M}_{\mathrm{c}}{}^{B_\mathrm{l}} + {}^{B}\vec{M}_{\mathrm{s}}{}^{\mathrm{SS_l}} + {}^{B}\vec{M}_{\mathrm{s}}{}^{\mathrm{PS_{ll}}} + {}^{B}\vec{M}_{\mathrm{s}}{}^{\mathrm{PS_{lt}}}, \end{aligned} \tag{2}$$

where the upper left superscript identifies the reference frame (F, track following, and B, body following) on which the forces are applied, and the right superscript identifies B_l , the leading bogie frame; $\mathrm{SS_l}$, the secondary suspension of the leading bogie frame; and $\mathrm{PS_{ll/lt}}$, the leading/trailing primary suspensions of the leading bogie frame. The right subscripts g, c, s refer instead to the gravity, centrifugal (not used for this work) and suspension forces.

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

Table 2. Nominal values of the suspension components, variances and TSI of the critical speed, obtained using the one-at-a-time analysis, the ANOVA expansion of the complete model and the more accurate ANOVA expansion of the reduced model.

Suspension	Nom. value	One-at-time	ANOVA		ANOVA-Ref.	
		$\bar{\sigma}_v$	$\bar{\sigma}_v$	TSI	$\bar{\sigma}_v$	TSI
PSLL_LEFT_K1	1823.0 kN/m	0.00	0.03	0.01		
PSLL_LEFT_K2	3646.0 kN/m	0.06	0.18	0.06	0.18	0.09
PSLL_LEFT_K3	3646.0 kN/m	0.02	0.13	0.04	0.14	0.07
PSLL_RIGHT_K1	1823.0 kN/m	0.00	0.05	0.02		
PSLL_RIGHT_K2	3646.0 kN/m	0.06	0.17	0.06	0.22	0.11
PSLL_RIGHT_K3	3646.0 kN/m	0.03	0.17	0.06	0.10	0.05
PSLT_LEFT_K1	1823.0 kN/m	0.00	0.02	0.01		
PSLT_LEFT_K2	3646.0 kN/m	0.54	1.71	0.56	1.29	0.63
PSLT_LEFT_K3	3646.0 kN/m	0.14	0.20	0.07	0.11	0.05
PSLT_RIGHT_K1	1823.0 kN/m	0.00	0.05	0.02		
PSLT_RIGHT_K2	3646.0 kN/m	0.55	1.73	0.56	1.22	0.59
PSLT_RIGHT_K3	3646.0 kN/m	0.03	0.13	0.04	0.17	0.08
SSL_LEFT_K4	182.3 kN/m	0.00	0.01	0.00		
SSL_LEFT_K5	333.3 kN/m	0.00	0.01	0.00		
SSL_LEFT_K6	903.35 kN/m	0.00	0.02	0.01		
SSL_LEFT_D1	20.0 kNs/m	0.00	0.02	0.01		
SSL_LEFT_D2	29.2 kNs/m	0.02	0.04	0.01		
SSL_LEFT_D6	166.67 kNs/m	0.00	0.02	0.01		
SSL_RIGHT_K4	182.3 kN/m	0.00	0.01	0.00		
SSL_RIGHT_K5	333.3 kN/m	0.00	0.00	0.00		
SSL_RIGHT_K6	903.35 kN/m	0.00	0.02	0.01		
SSL_RIGHT_D1	20.0 kNs/m	0.00	0.03	0.01		
SSL_RIGHT_D2	29.2 kNs/m	0.02	0.04	0.01		
SSL_RIGHT_D6	166.67 kNs/m	0.00	0.02	0.01		

Notes: The naming convention used for the suspensions works as follows. PSL and SSL stand for primary and secondary suspension of the leading bogie, respectively. The following L and T in the primary suspension stand for leading and trailing wheel sets. The last part of the nomenclature refers to the particular suspension components as shown in Figure 1.

The equations of motion for the wheel sets are given by

$$
\begin{aligned}
m\ddot{\vec{x}} &= {}^{F}\vec{F}_{\mathrm{g}}{}^{\mathrm{W_{II}}} + {}^{F}\vec{F}_{\mathrm{c}}{}^{\mathrm{W_{II}}} + {}^{F}\vec{F}_{\mathrm{L}}^{\mathrm{W_{II}}} + {}^{F}\vec{F}_{\mathrm{R}}^{\mathrm{W_{II}}} + {}^{F}\vec{F}_{\mathrm{s}}{}^{\mathrm{PS_{II}}} \\
J_\phi\ddot{\phi} &= \{{}^{B}M_{\mathrm{L}}^{\mathrm{W_{II}}}\}_\phi + \{{}^{B}M_{\mathrm{R}}^{\mathrm{W_{II}}}\}_\phi \\
&\quad + \{{}^{B}\vec{M}_{\mathrm{g}}{}^{\mathrm{W_{II}}}\}_\phi + \{{}^{B}\vec{M}_{\mathrm{c}}{}^{\mathrm{W_{II}}}\}_\phi + \{{}^{B}\vec{M}_{\mathrm{s}}{}^{\mathrm{PS_{II}}}\}_\phi \\
J_\chi\dot{\beta} &= \{{}^{B}M_{\mathrm{L}}^{\mathrm{W_{II}}}\}_\chi + \{{}^{B}M_{\mathrm{R}}^{\mathrm{W_{II}}}\}_\chi \\
J_\psi\ddot{\psi} &= \{{}^{B}M_{\mathrm{L}}^{\mathrm{W_{II}}}\}_\psi + \{{}^{B}M_{\mathrm{R}}^{\mathrm{W_{II}}}\}_\psi \\
&\quad + \{{}^{B}\vec{M}_{\mathrm{g}}{}^{\mathrm{W_{II}}}\}_\psi + \{{}^{B}\vec{M}_{\mathrm{c}}{}^{\mathrm{W_{II}}}\}_\psi + \{{}^{B}\vec{M}_{\mathrm{s}}{}^{\mathrm{PS_{II}}}\}_\psi,
\end{aligned}
\tag{3}
$$

where the same notation for Equation (2) was used, W stands for wheel set and the additional L and R subscripts indicate the left and right forces on the axle due to the wheel–rail contact forces. The pitch motion of the wheel set is substituted by the angular velocity perturbation β due to the odd distribution of the forces among the wheels.

The wheel–rail interaction is modelled using tabulated values generated with the routine RSGEO [7] for the static penetration at the contact points. These values are then updated using Kalker's [8] work for the additional penetrations. The creep forces are approximated using the Shen–Hedrick–Elkins nonlinear theory.[9] The complete deterministic system [6]

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

can be written abstractly as

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}(t) = \mathbf{f}(\mathbf{u}, t). \tag{4}$$

It is nonlinear, non-smooth and has 28 degrees of freedom.

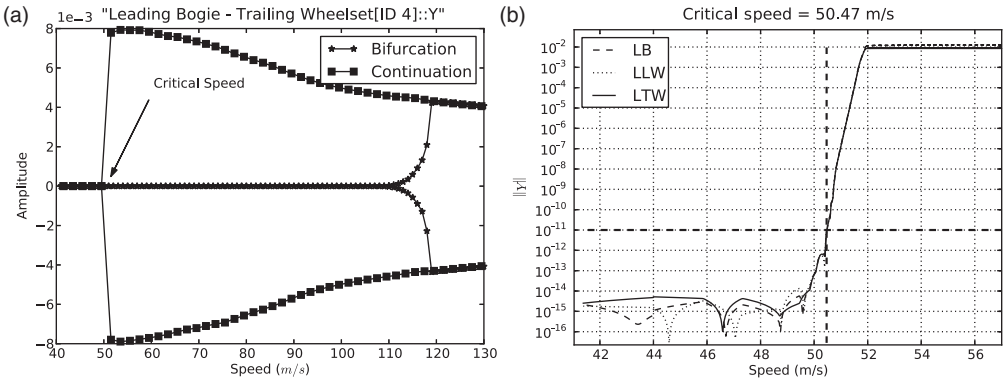
2.1. *Nonlinear dynamics of the deterministic model*

The deterministic dynamics of the complete wagon with a couple of Cooperrider bogies were analysed in [6]. The stability of the half-wagon model considered in this work is characterised by a sub-critical Hopf bifurcation at $v_{\mathrm{L}} = 114$ m/s, as it is shown in Figure 2(a), and a critical speed $v_{\mathrm{NL}} = 50.47$ m/s. The critical speed is found using a continuation method from the periodic limit cycle detected at a speed greater than the Hopf-bifurcation speed v_{L}. In order to save computational time, we try to detect the periodic limit cycle at speeds lower than v_{L} perturbing the system as described in [10]. This is the approach that we will take during all the computations of critical speeds in the next sections. The criterion used in order to detect the value of the critical speed is based on the power of the lateral oscillations in a 1 s sliding window of the computed solution. In particular, a threshold is selected – in this case a strict threshold of 10^{-11} was used – and the critical speed is defined as the speed at which the power of the lateral displacement of all the components fall below such threshold. Figure 2(b) shows how this criterion is applied.

2.2. *The stochastic model*

In the following, we will assume that the suspension characteristics are not deterministically known. Rather, they are described by probability distributions stemming from the manufacturing uncertainty or the wear.

If experimental information is available, then some standard distributions can be assumed and an optimisation problem can be solved in order to determine the statistical parameters of such distributions (e.g. mean, variance, etc.). Alternatively the probability density function (PDF) of the probability distribution can be estimated by kernel smoothing.[11, Ch. 6]



Bifurcation diagram Critical speed detection criteria

Figure 2. Left: complete bifurcation diagram where the folding point is detected by continuation (ramping) method from the periodic limit cycle. Right: criterion for the determination of the critical speed based on the power of the lateral oscillations in a sliding window. LB, LLW and LTW stand for the bogie frame, the leading wheel set and the trailing wheel set, respectively. (a) Bifurcation diagram and (b) critical speed detection criteria.

Due to the lack of data to the authors, in this work the probability distributions associated with the suspension components will be assumed to be Gaussian around their nominal value, with a standard deviation of 5%. We define $\mathbf{Z}$ to be the *d-dimensional* vector of random variables $\{z_i \sim \mathcal{N}(\mu_i, \sigma_i)\}_{i=1}^{d}$ describing the distributions of the suspension components, where d is called the co-dimension of the system. The stochastic dynamical system is then described by

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{u}(t, \mathbf{Z}) = \mathbf{f}(\mathbf{u}, t, \mathbf{Z}), \quad (0, T] \times \mathbb{R}^d. \tag{5}$$

3. Sensitivity analysis

Sensitivity analysis is used to describe how the model output depends on the input parameters. Such analysis enables the user to identify the most important parameters for the model output. Sensitivity analysis can be viewed as the search for the direction in the parameter space with the fastest growing perturbation from the nominal output.

One approach of sensitivity analysis is to investigate the partial derivatives of the output function with respect to the parameters in the vicinity of the nominal output. This approach goes by the name of local sensitivity analysis, stressing the fact that it works only for small perturbations of the system.

When statistical information regarding the parameters is known, it can be embedded in the global sensitivity analysis, which is not restricted to small perturbations of the system, but can handle bigger variability in the parameter space. This is the focus of this work and will be described in the following sections.

3.1. *Uncertainty quantification*

The solution of Equation (5) is $\mathbf{u}(t, \mathbf{Z})$, varying in the parameter space. The random vector $\mathbf{Z}$ is defined in the probability space $(\Omega, \mathcal{F}, \mu_{\mathbf{Z}})$, where $\mathcal{F}$ is the Borel set constructed on Ω and $\mu_{\mathbf{Z}}$ is a probability measure (i.e. $\mu_{\mathbf{Z}}(\Omega) = 1$). In UQ we are interested in computing the density function of the solution and/or its first moments, e.g. mean and variance:

$$\begin{aligned} \mu_{\mathbf{u}}(t) &= \mathbf{E}[\mathbf{u}(t, \mathbf{Z})]_{\rho_{\mathbf{Z}}} = \int_{\Omega^d} \mathbf{u}(t, \mathbf{z})\, \mathrm{d}F_{\mathbf{Z}}(\mathbf{z}), \\ \sigma_{\mathbf{u}}^2(t) &= \mathbf{Var}[\mathbf{u}(t, \mathbf{Z})]_{\rho_{\mathbf{Z}}} = \int_{\Omega^d} (\mathbf{u}(t, \mathbf{z}) - \mu_{\mathbf{u}}(t))^2\, \mathrm{d}F_{\mathbf{Z}}(\mathbf{z}), \end{aligned} \tag{6}$$

where $\rho_{\mathbf{Z}}(\mathbf{z})$ and $F_{\mathbf{Z}}(\mathbf{z})$ are the probability density function (PDF) and the cumulative distribution function (CDF), respectively. Several techniques are available to approximate these high-dimensional integrals. In the following, we present the two main classes of these methods.

3.1.1. *Sampling-based methods*

The most known sampling method is the Monte Carlo (MC) method, which is based on the law of large numbers. Its estimates are

$$\mu_{\mathbf{u}}(t) \approx \bar{\mu}_{\mathbf{u}}(t) = \frac{1}{M} \sum_{j=1}^{M} \mathbf{u}(t, \mathbf{Z}^{(j)}),$$

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

$$\sigma_{\mathbf{u}}^2(t) \approx \bar{\sigma}_{\mathbf{u}}^2(t) = \frac{1}{M-1}\sum_{j=1}^{M}(\mathbf{u}(t,\mathbf{Z}^{(j)}) - \bar{\mu}_{\mathbf{u}}(t))^2, \tag{7}$$

where $\{\mathbf{Z}^{(j)}\}_{j=1}^{M}$ are realisations sampled randomly with respect to the probability distribution $\mathbf{Z}$. The MC method has a probabilistic error of $\mathcal{O}(1/\sqrt{M})$, thus it suffers from the work effort required to compute accurate estimates (e.g. to improve an estimate of one decimal digit, the number of function evaluations necessary is 100 times bigger). However, the MC method is very robust because this convergence rate is independent of the co-dimensionality of the problem, so it is useful to get approximate estimates of very high-dimensional integrals.

Sampling methods with improved convergence rates have been developed, such as Latin hypercube sampling and quasi-MC methods. However, the improved convergence rate comes at the expense of several drawbacks, e.g. the convergence of quasi-MC methods is dependent on the co-dimensionality of the problem and Latin hypercube cannot be used for incremental sampling.

3.1.2. *Cubature rules*

The integrals in Equation (6) can also be computed using cubature rules. These rules are based on a polynomial approximation of the target function, i.e. the function describing the relation between parameters and QoI, so they have super-linear convergence rate on the set of smooth functions. Their applicability is, however, limited to low-co-dimensional problems because cubature rules based on a tensor grid suffer the *curse of dimensionality*, i.e. if m is the number of points used in the one-dimensional rule and d the dimension of the integral, the number of d points at which to evaluate the function grows as $\mathcal{O}(m^d)$. They will, however, be presented here because they represent a fundamental tool for the creation of high-dimensional model representations (HDMRs) that will be presented in the next section.

Let $\mathbf{Z}$ be a vector of *independent* random variables (i.e. $\mathbf{Z} : \Omega \to \mathbb{R}^d$) in the probability space $(\Omega, \mathcal{F}, \mu_{\mathbf{Z}})$, where $\mathcal{F}$ is the Borel set constructed on Ω and $\mu_{\mathbf{Z}}$ is the measure of $\mathbf{Z}$. By the independence of $\mathbf{Z}$, we can write Ω as a product space $\Omega = \times_{i=1}^{d}\Omega_i$, with product measure $\mu_{\mathbf{Z}} = \times_{i=1}^{d}\mu_i$. For $A \subseteq \mathbb{R}^d$, we call $F_{\mathbf{Z}}(A) = \mu_{\mathbf{Z}}(\mathbf{Z}^{-1}(A))$ the distribution of $\mathbf{Z}$.

For each independent dimension of Ω we can construct orthogonal polynomials $\{\phi_n(z_i)\}_{n=1}^{N_i}$, $i = 1,\ldots,d$, with respect to the probability distribution F_i, where $F_{\mathbf{Z}} = \times_{i=1}^{d}F_i$.[12] The tensor product of such basis forms a basis for

$$L^2_{F_{\mathbf{Z}}} \triangleq \left\{ f : I \subseteq \mathbb{R}^d \to \mathbb{R} \,\middle|\, \int_I f^2(\mathbf{z})\,\mathrm{d}F_{\mathbf{Z}}(\mathbf{z}) = \mathbf{Var}[f(\mathbf{Z})] < \infty \right\} \tag{8}$$

that means that there exists a projection operator $P_N : L^2_{F_{\mathbf{Z}}} \to \mathbb{P}^N$ such that for any $f \in L^2_{F_{\mathbf{Z}}}$, and with the notation $\mathbf{i} = (i_1,\ldots,i_d) \in [0,\ldots,N_1] \times \cdots \times [0,\ldots,N_d]$,

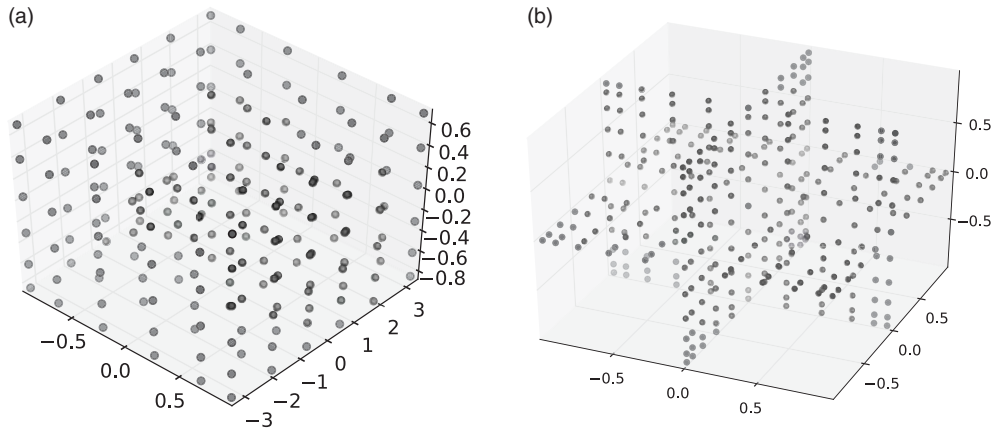
$$f \approx P_N f \triangleq \sum_{\mathbf{i}=0}^{N_1,\ldots,N_d} \hat{f}_{\mathbf{i}}\Phi_{\mathbf{i}}, \quad \hat{f}_{\mathbf{i}} \triangleq \frac{(f,\Phi_{\mathbf{i}})_{L^2_{F_{\mathbf{Z}}}}}{\|\Phi_{\mathbf{i}}\|^2_{L^2_{F_{\mathbf{Z}}}}}, \tag{9}$$

where $\Phi_{\mathbf{i}} = \prod_{k\in\mathbf{i}}\phi_k$, $\|f\|^2_{L^2_{F_{\mathbf{Z}}}} = (f,f)_{L^2_{F_{\mathbf{Z}}}}$ and

$$(f,g)_{L^2_{F_{\mathbf{Z}}}} = \int_{\mathbb{R}^d} f(\mathbf{z})g(\mathbf{z})\,\mathrm{d}F_{\mathbf{Z}}(\mathbf{z}). \tag{10}$$

In the following, we will be marginally interested in the approximation (9) of the QoI function. However, the fast – possibly spectral – convergence of such approximation is inherently

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014



Tensor grid cut-HDMR grid

Figure 3. Example of the distribution of points for tensor cubature rules (left) and the distribution of points for the cut-HDMR grid accounting for second-order interactions (right). (a) Tensor grid and (b) cut-HDMR grid.

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

connected with the convergence in the approximation of statistical moments, because $\mu_f = \hat{f}_{\mathbf{0}}$ and $\sigma_f^2 = \sum_{\mathbf{i}} \hat{f}_{\mathbf{i}}^2 - \hat{f}_{\mathbf{0}}^2$.[13]

From the orthogonal polynomials used in the construction of Equation (9), the one-dimensional Gauss quadrature points and weights $\{z_{j_i}, w_{j_i}\}_{j_i}^{N_i}$ can be derived using the Golub–Welsch algorithm.[12] Gauss quadrature points and weights $\{\mathbf{z}_{j_1,\dots,j_d}, w_{j_1,\dots,j_d}\}_{j_1,\dots,j_d=1}^{N_1,\dots,N_d}$ for the tensor product space can be obtained as tensor product of one-dimensional cubature rules (see Figure 3(a)), obtaining the following approximations for Equation (6):

$$\begin{aligned} \mu_{\mathbf{u}}(t) &\approx \bar{\mu}_{\mathbf{u}}(t) = \sum_{j_1}^{N_1} \cdots \sum_{j_d}^{N_d} \mathbf{u}(t, \mathbf{z}_{j_1,\dots,j_d}) w_{j_1,\dots,j_d}, \\ \sigma_{\mathbf{u}}^2(t) &\approx \bar{\sigma}_{\mathbf{u}}^2(t) = \sum_{j_1}^{N_1} \cdots \sum_{j_d}^{N_d} (\mathbf{u}(t, \mathbf{z}_{j_1,\dots,j_d}) - \bar{\mu}_{\mathbf{u}}(t))^2 w_{j_1,\dots,j_d}. \end{aligned} \tag{11}$$

Gauss quadrature rules of order N are accurate for polynomials of order up to degree $2N - 1$. This high accuracy comes at the expense of the curse of dimensionality due to the use of tensor products in high-dimensional integration. This effect can be alleviated by the use of sparse grid techniques proposed by Smolyak [14] that use an incomplete version of the tensor product. However, in the following section, we will see that we can often avoid working in very high-dimensional spaces.

3.2. *High-dimensional model representation*

High-dimensional models are very common in practical applications, where a number of parameters influence the dynamical behaviours of a system. These models are very difficult to handle, in particular if we consider them as black boxes where we are only allowed to change parameters. One method to circumvent these difficulties is the HDMR expansion,[15] where the high-dimensional function $f : \Omega \to \mathbb{R}$, $\Omega \subseteq \mathbb{R}^d$ is represented by a function decomposed

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

with lower order interactions:

$$f(\mathbf{x}) \equiv f_0 + \sum_i f_i(\mathbf{x}_i) + \sum_{i<j} f_{i,j}(\mathbf{x}_i, \mathbf{x}_j) + \cdots + f_{1,2,\ldots,d}(\mathbf{x}_1, \ldots, \mathbf{x}_d). \tag{12}$$

This expansion is exact and exists for any integrable and measurable function f, but is not unique. There is a rich variety of such expansions depending on the projection operator used to construct them. The most used in statistics is the ANOVA-HDMR where the low-dimensional functions are defined

$$\begin{aligned}
f_0^A &\equiv P_0^A f(\mathbf{x}) = \int_\Omega f(\mathbf{x})\, \mathrm{d}\mu(\mathbf{x}),\\
f_i^A(\mathbf{x}_i) &\equiv P_i^A f(\mathbf{x}) = \int_{\Omega_i} f(\mathbf{x}) \prod_{i\neq j} \mathrm{d}\mu_j(\mathbf{x}_j) - P_0^A f(\mathbf{x}),\\
f^A_{i_1,\ldots,i_l}(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l}) \equiv P^A_{i_1,\ldots,i_l} f(\mathbf{x}) &= \int_{\Omega_{i_1,\ldots,i_l}} f(\mathbf{x}) \prod_{\{k\notin i_1,\ldots,i_l\}} \mathrm{d}\mu_k(\mathbf{x}_k)\\
&\quad - \sum_{k_1<\cdots<k_{l-1}\in\{i_1,\ldots,i_l\}} P^A_{k_1,\ldots,k_{l-1}} f(\mathbf{x})\\
&\quad - \cdots - \sum_{k\in\{i_1,\ldots,i_l\}} P_k^A f(\mathbf{x}) - P_0^A f(\mathbf{x}),
\end{aligned} \tag{13}$$

where $\Omega_{i_1,\ldots,i_l} \subseteq \Omega$ is the hypercube excluding indices $i_1, \ldots, i_l$ and μ is the product measure $\mu(\mathbf{x}) = \prod_{i=1}^d \mu_i(\mathbf{x}_i)$. This expansion can be used to express the total variance of f, by noting that

$$\begin{aligned}
D \equiv \mathbf{E}[(f - f_0)^2] &= \sum_i D_i + \sum_{i<j} D_{i,j} + \cdots + D_{1,2,\ldots,d},\\
D_{i_1,\ldots,i_l} &= \int_{\Omega^{i_1,\ldots,i_l}} (f^A_{i_1,\ldots,i_l}(\mathbf{x}_{i_l}))^2 \prod_{k\in\{i_1,\ldots,i_l\}} \mathrm{d}\mu_k(\mathbf{x}_k),
\end{aligned} \tag{14}$$

where $\Omega^{i_1,\ldots,i_l} \subseteq \Omega$ is the hypercube including indices $i_1, \ldots, i_l$. However, the high-dimensional integrals in the ANOVA-HDMR expansion are computationally expensive to evaluate.

An alternative expansion is the cut-HDMR, which is built by superposition of hyperplanes passing through the cut centre $\mathbf{y} = (y_1, \ldots, y_d)$:

$$\begin{aligned}
f_0^C &\equiv P_0^C f(\mathbf{x}) = f(\mathbf{y}),\\
f_i^C(\mathbf{x}_i) &\equiv P_i^C f(\mathbf{x}) = f^i(\mathbf{x}_i) - P_0^C f(\mathbf{x}),\\
f^C_{i_1,\ldots,i_l}(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l}) \equiv P^C_{i_1,\ldots,i_l} f(\mathbf{x}) &= f^{i_1,\ldots,i_l}(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l})\\
&\quad - \sum_{k_1<\cdots<k_{l-1}\in\{i_1,\ldots,i_l\}} P^C_{k_1,\ldots,k_{l-1}} f(\mathbf{x})\\
&\quad - \cdots - \sum_{k\in\{i_1,\ldots,i_l\}} P_k^C f(\mathbf{x}) - P_0^C f(\mathbf{x}),
\end{aligned} \tag{15}$$

where $f^{i_1,\ldots,i_l}(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l})$ is the function $f(\mathbf{x})$ with all the remaining variables set to $\mathbf{y}$. This expansion requires the evaluation of the function f on lines, planes and hyperplanes passing through the cut centre.

If cut-HDMR (15) is a good approximation of f at order L , i.e. considering up to L-terms interactions in Equation (12), such expansion can be used for the computation of ANOVA-HDMR in place of the original function. This reduces the computational cost dramatically: let d be the number of parameters and s the number of samples taken along each direction (being them MC samples or cubature points), then the cost of constructing cut-HDMR in terms of function evaluations is

$$\sum_{i=0}^{L} \frac{\mathrm{d}!}{(\mathrm{d}-i)!i!}(s-1)^i. \tag{16}$$

3.3. *Total sensitivity indices*

The main task of sensitivity analysis is to quantify the sensitivity of the output with respect to the input. In particular, it is important to know how much of this sensitivity is accountable to a particular parameter. With the focus on global sensitivity analysis, the sensitivity of the system to a particular parameter can be expressed by the variance of the output associated with that particular input.

One approach to this question is to consider each parameter separately and to apply one of the UQ techniques introduced in Section 3.1. This approach goes by the name of one-at-a-time analysis. This technique is useful to get a first overview of the system. However, this technique lacks an analysis of the interaction between input parameters, which in many cases is important.

A better analysis can be achieved using the method of Sobol.[16] Here single sensitivity measures are given by

$$S_{i_1,\dots,i_\mathrm{l}} = \frac{D_{i_1,\dots,i_\mathrm{l}}}{D} \quad \text{for } 1 \le i_1 < \cdots < i_\mathrm{l} \le n, \tag{17}$$

where D and $D_{i_1,\dots,i_\mathrm{l}}$ are defined according to Equation (14). These express the amount of total variance that is accountable to a particular combination $i_1, \dots, i_\mathrm{l}$ of parameters. The TSI is the total contribution of a particular parameter to the total variance, including interactions with other parameters. It can be expressed by

$$\mathrm{TS}(i) = 1 - S_{\neg i}, \tag{18}$$

where $S_{\neg i}$ is the sum of all $S_{i_1,\dots,i_\mathrm{l}}$ that do not involve parameter i.

These TSIs can be approximated using sampling-based methods in order to evaluate the integrals involved in Equation (14). Alternatively, Gao and Hesthaven [4] suggest to use cut-HDMR and cubature rules in the following manner:

(1) compute the cut-HDMR expansion on cubature nodes for the input distributions (Figure 3(b)),
(2) derive the approximated ANOVA-HDMR expansion from the cut-HDMR,
(3) compute the TSI from the ANOVA-HDMR.

This approach gives the freedom of selecting the level of accuracy for the HDMR expansion depending on the level of interaction between parameters. The truncation order L of the ANOVA-HDMR can be selected and the accuracy of such expansion can be assessed using the concept of 'effective dimension' of the system: for $q \le 1$, the effective dimension of the

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

integrand f is an integer L such that

$$\sum_{0<|t|\leq L} D_t \geq qD, \tag{19}$$

where t is a multi-index $i_1, \ldots, i_l$ and $|t|$ is the cardinality of such multi-index. The parameter q is chosen based on a compromise between accuracy and computational cost.

4. Sensitivity analysis on railway vehicle dynamics

The study of uncertainty propagation and sensitivity analysis through dynamical systems is a computationally expensive task. In this analysis, we adopt a collocation approach, where we study the behaviours of ensembles of realisations. From the algorithmic point of view, the quality of a method is measured in the number of realisations needed in order to infer the same accuracy in statistics. Each realisation is the result of an initial value problem (IVP) computed using the program DYnamics Train SImulation developed in [6], where the model presented in Section 2 has been set up and the IVP has been solved using the explicit Runge–Kutta–Fehlberg method ERKF34.[17] An explicit solver has been used in light of the analysis performed in [18], where it was found that the hunting motion could be missed by implicit solvers, used with relaxed tolerances, due to numerical damping. In particular, implicit solvers are frequently used for stiff problems, like the one treated here, because their step size is bounded by accuracy constraints instead of stability. However, the detection of the hunting motion requires the selection of strict tolerances, reducing the allowable step sizes and making the implicit methods more expensive than the explicit ones. Since the collocation approach for UQ involves the computation of completely independent realisations, this allows for a straightforward parallelisation of the computations on clusters. Thus, 25 nodes of the DTU cluster have been used to speed up the following analysis. The first step in the analysis of a stochastic system is the characterisation of the probability distribution of the QoI. Since the complete model has co-dimension 24, a traditional sampling method, among the ones presented in Section 2, is the most suited for the task of approximating the integrals in Equation (6). Figure 4(a) shows the histogram of the computed critical speeds with respect to the uncertainty in the suspension components. In order to speed up the convergence, we used 200 samples generated with the Latin hypercube method.[19] Kernel smoothing [11] has been used to estimate the density function according to this histogram. The estimated mean and variance are $\bar{\mu}_v = 51.83$ m/s and $\bar{\sigma}_v = 4.07\,\mathrm{m}^2/\mathrm{s}^2$. It is important to keep in mind that the first two moments do not account for all the information about the distribution of the QoI unless it is Gaussian. As shown in Figure 4(a), the distribution is not Gaussian and the ensemble spans approximately 14 m/s. However, in this particular case, the outliers appear only in the upper end of the distribution, whereas the lower end is fairly well defined by the ensemble.

4.1. *One-at-a-time analysis*

When each suspension component is considered independently from the others, the estimation problem in Equation (6) is reduced to the calculation of an one-dimensional integral. This task can be readily achieved by quadrature rules that have proven to be computationally more efficient on problems of this dimensionality than sampling methods.[3] Fourth-order quadrature rules have been used to approximate the variances due to the single components. The

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

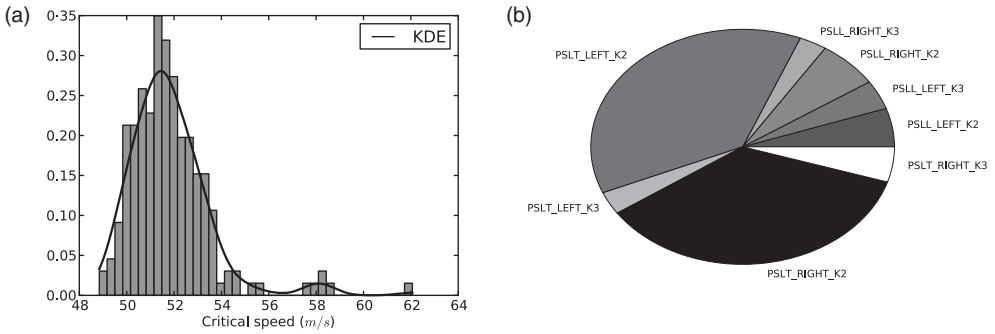


Figure 4. Left: histogram of the critical speed obtained using Latin hypercube sampling and the estimated density function (KDE) obtained using kernel smoothing. Right: pie plot of the TSI on the reduced stochastic model, where only the most influential components are analysed (see Table 2 for an explanation of the notation).

convergence of this method enables a check of accuracy through the decay of the expansion coefficients of the target function.[3]

The second column in Table 2 lists the results of such analysis. The amount of variance described by this analysis is given by the sum of all the variances: $\hat{\sigma} = 1.47\,\mathrm{m}^2/\mathrm{s}^2$. This quantity is far from representing the total variance of the stochastic system, suggesting that interactions between suspension components are important. Anyway the method is useful to make a first guess about which components are the most important: the critical speed of the railway vehicle model analysed in this work shows a strong sensitivity related to the longitudinal springs (K2) in the trailing wheel set.

4.2. *Total sensitivity analysis*

The technique outlined in Section 3.3 can fulfil three important tasks: taking into account parameter interactions, performing the analysis with a limited number of realisations and enabling an error control in the approximation. In a first stage, we consider the full stochastic model and construct a cut-HDMR expansion which takes into account second-order interactions and describes the target function through second-order polynomials, computing the realisations for up to two-dimensional cubature rules. The ANOVA-HDMR expansion of the cut-HDMR expansion can be quickly computed, due to the low dimensionality of the single terms in Equation (15). At this point, the $D_{i_1,\dots,i_l}$ values in Equation (14) can be obtained and the effective dimensionality of the target function, given by Equation (19) for $q = 0.95$, is found to be $L = 2$. This confirms that the first- and second-order interactions are sufficient to describe most of the variance. The third and fourth columns of Table 2 list the total variances induced by each parameter, including interactions with other parameters, and the Sobol TSIs.

Once the first approximation of the sensitivities is obtained, the parameters with the lowest sensitivity indices can be fixed to their nominal values and we can perform a more accurate analysis of the remaining stochastic system. Longitudinal and vertical springs (K2 and K3) in the primary suspensions have shown to be very influential for the critical speed of the analysed model, thus a new cut-HDMR expansion, with second-order interactions and fourth-order polynomial approximation, is constructed. The resulting total variances and TSI are listed in the fifth and sixth columns of Table 2. A visual representation of the sensitivity indices is shown in the pie chart in Figure 4(b).

The results obtained by the one-at-a-time analysis are confirmed here by the total sensitivity analysis, but we stress that the latter provide a higher reliability because they describe a bigger part of the total variance of the complete stochastic system.

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

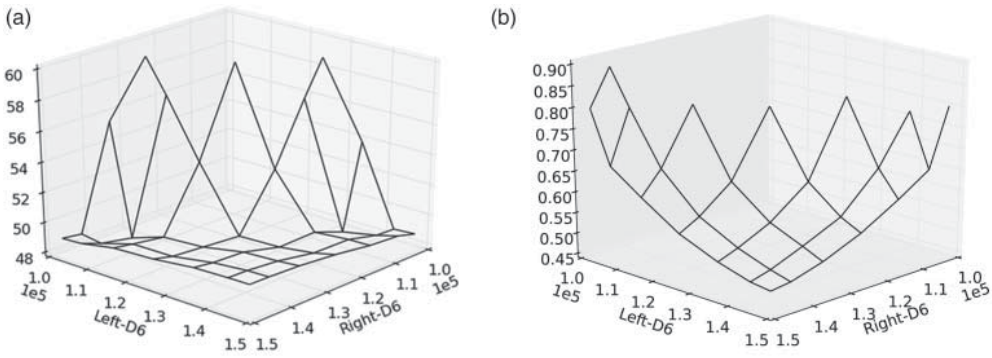


Figure 5. Critical speed and maximum Nadal's ratio with respect to the yaw damping coefficients on the left and right side of the bogie frame. We can see that the value of the critical speed is not significantly affected by the value of the yaw damping coefficient for the mean value chosen for sensitivity analysis (1.66×10^5 Ns/m). However, if the yaw damping coefficient is lowered too much, the intensity of the lateral oscillations increase, as shown by the growing Nadal ratio. The missing values in the critical speed plot are due to the oscillations being so big, that the model exit the computational domain for which the employed contact model works. The missing values in the Nadal's ratio plot are both due to the computations exiting the domain of the contact model and due to the vertical force being zero (lifting) at some instants during the ramping of the speed for the computation of the critical speed. (a) Critical speed vs. D6 and (b) maximum Nadal's ratio vs. D6.

4.3. *Discussion of the obtained results*

Even if the results obtained are formally correct, the interpretation of such results can raise some questions. A railway engineer might wonder why the yaw dampers D6 are not listed among the most important by the sensitivity analysis. The yaw dampers in the secondary suspension are known to provide stability to the vehicle ride, helping to increase its critical speed. This result is true also with the vehicle model considered here, in fact low values of D6 cause a drastic worsening of the ride stability. However, the TSI embed the probability distributions of the uncertain parameters in the global sensitivity analysis: the impact of a component is weighted according to these distributions. Thus, we say that the yaw damper has little influence on the riding stability with respect to the distributions chosen. A change in the distributions can dramatically change these results, thus *particular care should be taken with the quantification of the source of uncertainty*. To better show this fact, we looked for the relation of the critical speed with respect to the yaw dampers, for values below the mean value used for sensitivity analysis (1.66×10^5 Ns/m). We selected a range between $[1.0 \times 10^5, 1.5 \times 10^5]$ Ns/m and looked at the value of the critical speed. Figure 5(a) shows such response surface: the critical speed is not significantly changing when the yaw damping is high, as it is the case for the nominal value used in sensitivity analysis, but it increases drastically when the yaw damping is lowered too much. Unfortunately this does not mean that the car will run more safely. On the contrary, Figure 5(b) shows that the maximum Nadal's ratio, obtained while decreasing the speed in the continuation method for the detection of the critical speed, increases while lowering the yaw damping parameters. This suggests that the lateral oscillations become more violent and less compensated by the vertical forces. The missing values in Figure 5(a) and 5(b) are due to the lateral oscillations being outside the range of applicability of the contact model employed. Additionally, Figure 5(b) has some missing values due to the lifting of a wheel, leading to zero vertical forces.

This example suggests some observations on the extent to which sensitivity analysis should be used: it provides a measure of how much a QoI depends on a parameter, when the parameter value is not exactly known. In principle, from a risk management perspective, we would

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

like the QoI not to be sensitive to any parameter – i.e. the change in QoI should be little with respect to the parameter, like the yaw damper in the flat part of Figure 5(a). The fact that a QoI is sensitive to a certain parameter does not mean that this will be dangerous, but it must lead to a more detailed investigation. Furthermore, in real cases of virtual homologation we must look at several QoIs, as the previous example showed for the critical speed and the maximum Nadal's ratio.

4.4. *Remarks on UQ and sensitivity analysis*

The first question that an engineer performing analysis of a stochastic model has to wonder about is whether the uncertain input parameters considered are independent from a probabilistic point of view (we remind that the events A, B are independent if $\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B)$) or at least uncorrelated. In motivating our example of the uncertainty on the suspension components, we mentioned that their values are uncertain at the manufacturing time and are even more uncertain after thousands of running kilometres, due to the wear. However, the two cases are slightly different: in the first case, the value of each component can be considered independent and uncorrelated from the others, whereas in the second case the wear on each of the components cannot be considered independent from the others, because they undergo coupling dynamics. This does not mean we can do nothing, but we need first to find a map from the correlated random variables, to some lower dimensional uncorrelated random variables. If the distributions are Gaussians, a simple Cholesky factorisation of the correlation matrix will be sufficient as a map. In this case uncorrelation implies independency and we are well set for the application of the methods presented. If the distributions are non-Gaussian, then additional care should be paid to the particular problem at hand and one possible solution is the application of the Rosenblatt transformation.[13]

The second remark regards the influence of the selection of the QoI in UQ and sensitivity analysis. In Section 2.1, the continuation method used to estimate the critical speed was presented and the threshold used to determine the end of the hunting motion was chosen in a conservative way, as it is shown in Figure 2(b). Thus, the value of the computed critical speed will depend also on the deceleration chosen for the continuation method, i.e. the computed critical speed will be exact in the limit when the deceleration goes to zero. Of course, the exact computation of the critical speed is not computationally feasible. With the limited computational resources available, we then chose a fixed deceleration coefficient for the continuation method, and thus we introduced numerical uncertainty in the computations. Furthermore, the value has been found to be numerically accurate up to the first decimal digit, due to different choices of initial conditions and the tolerances set in the time steppers (these can have a large effect, considering the long-time integration needed for this problem and the accumulation of rounding errors). Therefore, the variance expressed from the analysis is given both by the variance due to the stochastic system and the variance introduced by the computation of the QoI. This is, however, a conservative consequence, meaning that a decision taken on the basis of the computed results is at least as safe as a decision taken using the 'exact results'. A test performed with different initial conditions showed that the sensitivity values found are qualitatively accurate up to the first decimal digit.

5. Conclusions

Sensitivity analysis is of critical importance in a wide range of engineering applications. The traditional approach of local sensitivity analysis is useful in order to characterise the

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

behaviour of a dynamical system in the vicinity of the nominal values of its parameters, but it fails in describing wider ranges of variations, e.g. caused by long-term wear. The global sensitivity analysis aims at representing these bigger variations and at the same time it embeds the probability distributions of the parameters in the analysis. This enables the engineer to take decisions, such as improving a design, based on the partial knowledge of the system.

Wrongly approached, global sensitivity analysis can turn to be a computationally expensive or even prohibitive task. In this work, a collection of techniques are used in order to accelerate such analysis for a high-co-dimensional problem. Each of the techniques used allows for a control of the accuracy, e.g. in terms of convergence rate for the cubature rules in Section 3.1 and the 'effective dimension' in Section 3.3. This makes the framework flexible and easy to be adapted to problems with more diversified distributions and target functions.

The analysis performed on the half-wagon equipped with a Cooperrider bogie shows a high importance of the longitudinal primary suspensions, and this reflects the connection between hunting and yaw motion.

It is important to notice that the same settings for global sensitivity analysis can be used for the investigation of different QoIs, such as wear in curved tracks, angle of attack, etc. once they have been properly defined. Furthermore, the 'non-intrusive' approach taken allows the engineer to use closed software for the computations. The machinery for sensitivity analysis needs only to be wrapped around it, without additional implementation efforts.

Downloaded by [DTU Library], [Daniele Bigoni] at 02:08 08 May 2014

References

[1] Funfschilling C, Perrin G, Kraft S. Propagation of variability in railway dynamic simulations: application to virtual homologation. Veh Syst Dyn. 2012;50:245–261.

[2] Mazzola L, Bruni S. Effect of suspension parameter uncertainty on the dynamic behaviour of railway vehicles. Appl Mech Mater. 2011;104:177–185.

[3] Bigoni D, Engsig-Karup A, True H. Comparison of classical and modern uncertainty quantification methods for the calculation of critical speeds in railway vehicle dynamics. 13th Mini Conference on Vehicle System Dynamics, Identification and Anomalies, Budapest, Hungary; 2012.

[4] Gao Z, Hesthaven J. Efficient solution of ordinary differential equations with high-dimensional parametrized uncertainty. Comm Comput Phys. 2011;10:253–286.

[5] Cooperrider N. The hunting behavior of conventional railway trucks. ASME J Eng Ind. 1972;94:752–762.

[6] Bigoni D. Curving dynamics in high speed trains [dissertation]. Kongens Lyngby: The Technical University of Denmark; 2011.

[7] Kik W, Moelle D. ACRadSchiene – to create or approximate wheel/rail profiles – tutorial; 2007. Available from: http://www.argecare.com/RSPROG.htm

[8] Kalker J. Wheel–rail rolling contact theory. Wear. 1991;144:243–261.

[9] Shen Z, Hedrick J, Elkins J. A comparison of alternative creep-force models for rail vehicle dynamic analysis. 8th IAVSD Symposium, Cambridge, UK; 1984.

[10] True H. Multiple attractors and critical parameters and how to find them numerically: the right, the wrong and the gambling way. Veh Syst Dyn. 2013;51(3):443–459.

[11] Friedman J, Hastie T, Tibshirani R. The elements of statistical learning. Vol. 1, Springer Series in Statistics. New York: Springer Verlag; 2001.

[12] Gautschi W. Orthogonal polynomials: computation and approximation. Numerical Mathematics and Scientific Computation. Oxford: Oxford University Press; 2004.

[13] Xiu D. Numerical methods for stochastic computations: a spectral method approach. Princeton (NJ): Princeton University Press; 2010.

[14] Petras K. Smolyak cubature of given polynomial degree with few nodes for increasing dimension. Numer Math. 2003;93:729–753.

[15] Rabitz H, Ömer FA. Sensitivity analysis. Chichester: Wiley; 2000. p. 199–223.

[16] Chan K, Tarantola S, Saltelli A, SobolÍM. Sensitivity analysis. Chichester: Wiley. p. 168–197.

[17] Hairer E, Norsett SP, Wanner G. Solving ordinary differential equations I: nonstiff problems. 2nd revision ed. Springer Series in Computational Mathematics. Berlin: Springer-Verlag; 1991.

[18] True H, Engsig-Karup A, Bigoni D. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics. Math Comput Simul. 2012;95:78–97.

[19] Mckay M, Beckman R, Conover W. A comparison of three methods for selecting values of input variables in the analysis of output from a comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics. 2000;41:55–61.

Proceedings of the ASME 2014 12th Biennial Conference on Engineering Systems Design and Analysis
ESDA2014
June 25-27, 2014, Copenhagen, Denmark

ESDA2014-20529

GLOBAL SENSITIVITY ANALYSIS OF RAILWAY VEHICLE DYNAMICS ON CURVED TRACKS

Daniele Bigoni
Dept. of Applied Mathematics and Computer Science
The Technical University of Denmark
Kgs.Lyngby, Denmark DK-2800
Email: dabi@dtu.dk

Allan P. Engsig-Karup
Hans True
Dept. of Applied Mathematics and Computer Science
The Technical University of Denmark
Kgs.Lyngby, Denmark DK-2800

ABSTRACT

This work addresses the problem of the reliability of simulations for realistic nonlinear systems, by using efficient techniques for the analysis of the propagation of the uncertainties of the model parameters through the dynamics of the system. We present the sensitivity analysis of the critical speed of a railway vehicle with respect to its suspension design. The variance that stems from parameter tolerances of the suspension is taken into account and its propagation through the dynamics of a full car with a couple of two-axle Cooperrider bogies running on curved track is studied.
Modern Uncertainty Quantification methods, such as Stochastic Collocation and Latin Hypercube, are employed in order to assess the global uncertainty in the computation of the critical speed. The sensitivity analysis of the critical speed to each parameter and combination of parameters is then carried out in order to quantify the importance of different suspension components. This is achieved using combined approaches of sampling methods, ANOVA expansions, Total Sensitivity Indices and Low-dimensional Cubature Rules.

NOMENCLATURE

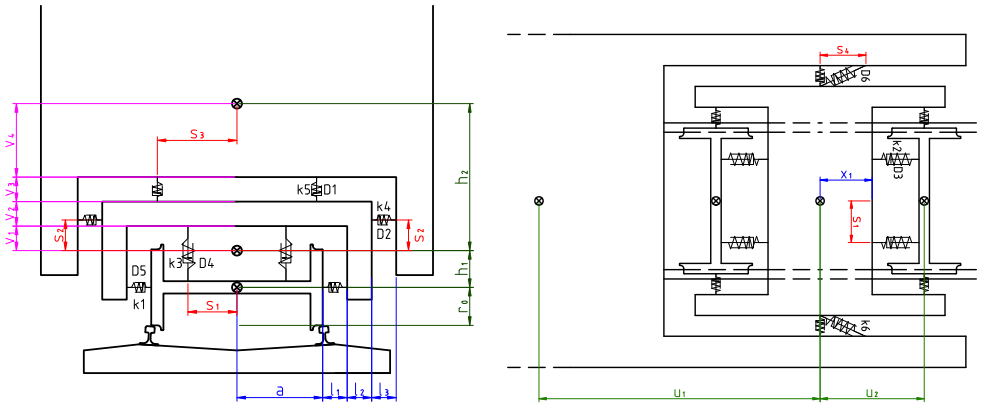
SSL/T Leading/Trailing Secondary Suspensions
PS Primary Suspensions
LL Leading wheel set on the Leading bogie frame
LT Trailing wheel set on the Leading bogie frame
TL Leading wheel set on the Trailing bogie frame
TT Trailing wheel set on the Trailing bogie frame

INTRODUCTION

The last couple of decades have seen the advent of Computer-Aided Design in many areas of engineering. This allows for enhanced design capabilities and the prediction and understanding of dangerous phenomena that would be difficult and expensive to reproduce in physical experiments. The simulation of deterministic physical systems, however, falls short in the task of explaining the phenomena that happen in reality. One part of the problem comes from the fact that models by definition are simplification of the reality and the engineer in charge of making a model bears always in mind Einstein's words: "Everything should be made as simple as possible, but not simpler". This part of uncertainty is very difficult to be dealt with and the validity of a particular model can be assessed only through experimentation. A second kind of uncertainty is related to the correctness of the working conditions at which the model is applied: in this case the model is assumed to be describing the physics accurately, but its working conditions – the parameters involved in the model – don't match the reality. This kind of parametric uncertainty can be dealt with and the continuous improvements in computational science allows for more involved analysis of the uncertainty.

In this work we will deal with the safety analysis of a complete rail car running on curved track. Railway vehicle dynam-

Copyright © 2014 by ASME



(a) Front view (b) Top view

FIGURE 1: THE RAIL CAR.

ics is subject to a number of uncertainties that can affect the rider's safety. Some of them are external loads applied to the system, such as track perturbations, wind gusts or different dispositions of the loaded goods. Others uncertainties are related to the car design, such as the suspension characteristics and the wheel wear.

The work will focus on the stability of a rail car equipped with two Cooperrider bogies and running on a curved track [1] under uncertain suspension characteristics, due to manufacturing tolerances. It is now well known that railway vehicles running at speeds higher than a fixed critical speed develop what is called the hunting motion: a sideways periodic or chaotic oscillation that can lead to increased wheel-rail wear and worsened comfort. This phenomenon can be described in terms of nonlinear dynamics of the system [2] and analyzed using suitable numerical methods for non-smooth dynamical systems [3].

The analysis of the uncertainty of the riding safety of the vehicle model will not be limited to the quantification of the total uncertainty, but will also focus on the identification of the parameters that most influence it. We will do it from a probabilistic point of view, where the safety is more or less sensitive to a particular suspension component depending on how much of the uncertainty is caused by it. This allows the engineer to detect the critical components that are required to be very accurately built by the manufacturer.

THE VEHICLE MODEL

The vehicle model chosen for this work is a complete rail car equipped with two Cooperrider bogies and four axles with wheel profile S1002, running on a curved track with rail profile UIC60. The rails have a cant of $1/40$. A total of 48 suspension components connect the car body, the bogie frames and the wheel-sets. Figure 1 shows the top and frontal view of half of the car. The dimensions, the masses and the inertia values of the components of the car are listed in Tab. 1, where the subscript c stands for car body, f for bogie frame, w for wheel set. The dynamical system is described using the Newton-Euler formulation:

$$\begin{aligned} \sum_{i=1}^{n} \vec{F}_i &= m\vec{a}, \\ \sum_{i=1}^{m} M_i &= \frac{d}{dt}\left([J]\cdot\vec{\omega}\right) + \vec{\omega}\times\left([J]\cdot\vec{\omega}\right), \end{aligned} \tag{1}$$

where $\vec{F}_i$ and $\vec{M}_i$ are the forces and torques applied on the center of mass of the bodies, m and $[J]$ are the mass and tensor moment of inertia respectively, $\vec{a}$ and $\vec{\omega}$ are the linear acceleration and the angular acceleration of the bodies.

In our model we will neglect the longitudinal displacements because we will not take into account the brake and the acceleration of the car. We will consider lateral and vertical displacement for all the bodies in the car and we will account also for their three possible rotations. On the wheel set the pitch angle will not be considered and instead we will consider only its angular velocity, to describe the rotation of the wheels. This results in a system of 66 coupled ordinary differential equations (ODEs) describing 35 degrees of freedom.

The static penetration at the contact points between wheels and rails is obtained using the routine RSGEO [4]. These values are tabulated and interpolated as needed during the solution of the system of ODEs and updated according to Kalker's work [5] in

Copyright © 2014 by ASME

Parm.	Value	Unit	Parm.	Value	Unit
r_0	0.425	$[m]$	a	0.75	$[m]$
h_1	0.0762	$[m]$	h_2	1.5584	$[m]$
l_1	0.30	$[m]$	l_2	0.30	$[m]$
l_3	0.30	$[m]$	x_1	0.349	$[m]$
v_1	0.6488	$[m]$	v_2	0.30	$[m]$
v_3	0.30	$[m]$	v_4	0.3096	$[m]$
s_1	0.62	$[m]$	s_2	0.6584	$[m]$
s_3	0.68	$[m]$	s_4	0.759	$[m]$
u_1	7.5	$[m]$	u_2	1.074	$[m]$
m_c	44388.0	$[kg]$	I_{cx}	$2.80 \cdot 10^5$	$[kgm^2]$
I_{cy}	$5.0 \cdot 10^5$	$[kgm^2]$	I_{cz}	$5.0 \cdot 10^5$	$[kgm^2]$
m_f	2918.0	$[kg]$	I_{fx}	6780.0	$[kgm^2]$
I_{fy}	6780.0	$[kgm^2]$	I_{fz}	6780.0	$[kgm^2]$
m_w	1022.0	$[kg]$	I_{wx}	678.0	$[kgm^2]$
I_{wy}	80.0	$[kgm^2]$	I_{wz}	678.0	$[kgm^2]$
K1	1823.0	$[kN/m]$	K2	3646.0	$[kN/m]$
K3	3646.0	$[kN/m]$	K4	182.3	$[kN/m]$
K5	333.3	$[kN/m]$	K6	903.35	$[kN/m]$
D1	20.0	$[kNs/m]$	D2	29.2	$[kNs/m]$
D6	166.67	$[kNs/m]$			

TABLE 1: DIMENSIONS, MASS, INERTIA AND SUSPENSION PARAMETERS OF THE RAIL CAR.

order to account for the additional penetration due to the dynamics. The creep forces are approximated using the Shen-Hedrick-Elkins nonlinear theory [6].
The complete deterministic system is nonlinear and non-smooth and can be written abstractly as

$$\frac{d}{dt}\mathbf{u}(t) = \mathbf{f}(\mathbf{u}, t). \tag{2}$$

The model is implemented in a general framework [1] for the simulation of railway vehicle dynamics on tangent or curved tracks. The framework allows, among other things, to select a variety of numerical ODE solvers and perform some analysis of the nonlinear dynamics of the system.

Nonlinear dynamics of the Deterministic Model

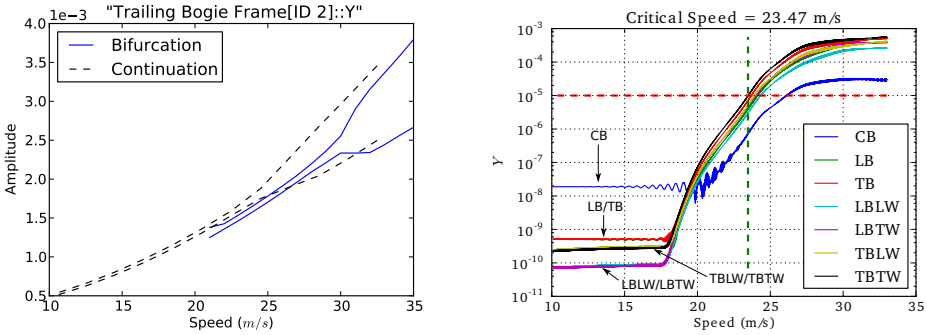
The dynamics of the complete car presented in the previous section were analyzed in [1], for trains running on tangent and curved tracks. On tangent tracks the car undergoes a sub-critical Hopf-bifurcation at a speed of $v_L = 114m/s$, entering a periodic limit cycle. This sub-critical Hopf-bifurcation is characterized by a significant fold, setting the critical speed of the car to $v_{NL} = 50.47m/s$. On tangent tracks the Hopf-bifurcation can be found using the Lyapunov's second method for stability and exploiting the fact that the center line of the track is a point of equilibrium for the system. The critical speed is then found using a continuation method following the periodic limit cycle backward (i.e. decreasing the speed quasi-statically).
On curved track, the Lyapunov's second method cannot be used anymore because the center line is not a point of equilibrium anymore. Thus the system of ODEs needs to be solved first accelerating, to detect the Hopf-bifurcation, and then decelerating to detect the critical speed for the curve under analysis. It is well known now that the critical speed may decrease when the train is running through a curve rather than on tangent track. Furthermore it was found that for some combination of curve profile and vehicle model, the sub-critical Hopf-bifurcation merges with the fold into a super-critical Hopf-bifurcation: this means that the speed where the Hopf-bifurcation occurs is also the one where the periodic limit cycle (the hunting motion) disappears when ramping down the velocity.

Figure 2 shows an example of a bifurcation analysis for the car running through a curve with radius $1600m$ and with the track super-elevated on the outer rail of $110mm$. Both the bifurcation point and the folding point cannot be detected precisely, but we can design a criteria based on the qualitative observation of the data. Using a sliding window Fourier analysis of the lateral displacement of the different components, and adjusting for the fact that the train is running on a curved track, we can define a detection criteria for the end of the hunting motion, based on the remaining power in the signal $\|Y\|$: a threshold of 10^{-5} was found to be a good indicator of the disappearance of the hunting motion. The application of such criteria can be seen in Fig. 2b. The legend in the figure stands for the different bodies: `CB`="Car Body", `LB`="Leading Bogie frame", `TB`="Trailing Bogie Frame", `LBLW`="Leading Wheel-set of the Leading Bogie frame", and so on.

The Stochastic Model

In the previous model we made the unrealistic assumption that we knew exactly the parameters involved in the system. From now on we will admit that the suspension parameters are not exactly known, but we can describe them with probability distributions. With this setting we want to model the realistic case where manufacturing fluctuations are present in the suspen-

Copyright © 2014 by ASME



(a) Bifurcation diagram

(b) Critical speed detection criteria

FIGURE 2: NONLINEAR DYNAMICS OF THE RAIL CAR ON CURVED TRACK.

sion components.

In a rigorous setting, the distribution of such parameters should be assessed from collected data. Several approaches, that make different assumptions, are available in order to construct a probability distribution from data. One of the most popular is the Kernel Smoothing [7, Ch. 6].

Due to the lack of data, in this work the probability distributions of the suspension parameters will be assumed to be Gaussian around their nominal values with a standard deviation of 5%. This assumption does not undermine the applicability of the method to other settings, where other distributions might be more suitable. We let $\mathbf{Z}$ be the *d-dimensional* vector of random variables $\{z_i \sim \mathcal{N}(\mu_i, \sigma_i)\}_{i=1}^{d}$ describing the suspension parameters, where d is called the *co-dimension* of the system. The stochastic dynamical system that we will aim to solve is then of the form

$$\frac{d}{dt}\mathbf{u}(t,\mathbf{Z}) = \mathbf{f}(\mathbf{u},t,\mathbf{Z}), \qquad (0;T]\times R^d. \tag{3}$$

With this system we will investigate the critical speed $v_{NL}(\mathbf{Z})$ and the sensitivity of it with respect to $\mathbf{Z}$.

SENSITIVITY ANALYSIS

Sensitivity analysis is used to identify the input parameters that affect the model output in the biggest amount. This analysis provides a useful tool to engineers in both the design phase and in the risk analysis phase of the production.

The traditional approach to a sensitivity analysis is to investigate the partial derivatives of a Quantity of Interest (QoI) with respect to the parameters. The directions with the highest gradients will be considered the most influential. Due to the locality of derivatives, this method goes under the name of local sensitivity analysis and it reduces to the computation of finite difference formulas around the nominal values of the parameters.

In this work we will instead look at the *global sensitivity*: the most influential parameters in the system are represented by the ones that give the biggest contribution to the total variance of the model output. This approach is not restricted to small perturbations, but it takes into account the uncertainty on the parameter values.

Uncertainty Quantification (UQ)

The solution of (3) is $\mathbf{u}(t,\mathbf{Z})$, varying in the parameter space. The random vector $\mathbf{Z}$ is defined in the probability space $(\Omega, \mathscr{F}, \mu_{\mathbf{Z}})$, where $\mathscr{F}$ is the Borel set constructed on Ω and $\mu_{\mathbf{Z}}$ is a probability measure (i.e. $\mu_{\mathbf{Z}}(\Omega) = 1$). In uncertainty quantification we are interested in computing the density function of the solution and/or its first moments, e.g. mean and variance:

$$\begin{aligned} \mu_{\mathbf{u}}(t) &= \mathbf{E}\left[\mathbf{u}(t,\mathbf{Z})\right]_{\rho_{\mathbf{Z}}} = \int_{\Omega^d} \mathbf{u}(t,\mathbf{z}) dF_{\mathbf{Z}}(\mathbf{z}), \\ \sigma^2_{\mathbf{u}}(t) &= \mathbf{Var}\left[\mathbf{u}(t,\mathbf{Z})\right]_{\rho_{\mathbf{Z}}} = \int_{\Omega^d} \left(\mathbf{u}(t,\mathbf{z}) - \mu_{\mathbf{u}}(t)\right)^2 dF_{\mathbf{Z}}(\mathbf{z}), \end{aligned} \tag{4}$$

where $\rho_{\mathbf{Z}}(\mathbf{z})$ and $F_{\mathbf{Z}}(\mathbf{z})$ are the probability density function (PDF) and the cumulative distribution function (CDF) respectively. Several techniques are available to approximate these high-dimensional integrals. In the following we present the two main classes of these methods.

Sampling based methods. The most known sampling method is the Monte Carlo (MC) method, which is based on the

Copyright © 2014 by ASME

law of large numbers. Its estimates are:

$$\begin{aligned}\mu_{\mathbf{u}}(t) &\approx \bar{\mu}_{\mathbf{u}}(t) = \frac{1}{M}\sum_{j=1}^{M}\mathbf{u}\left(t,\mathbf{Z}^{(j)}\right),\\ \sigma^2_{\mathbf{u}}(t) &\approx \bar{\sigma}^2_{\mathbf{u}}(t) = \frac{1}{M-1}\sum_{j=1}^{M}\left(\mathbf{u}\left(t,\mathbf{Z}^{(j)}\right)-\bar{\mu}_{\mathbf{u}}(t)\right)^2,\end{aligned} \tag{5}$$

where $\left\{\mathbf{Z}^{(j)}\right\}_{j=1}^{M}$ are realizations sampled randomly with respect to the probability distribution $\mathbf{Z}$. The MC method has a probabilistic error of $\mathscr{O}(1/\sqrt{M})$, thus it suffers from the work effort required to compute accurate estimates (e.g. to improve an estimate of one decimal digit, the number of function evaluations necessary is 100 times bigger). However the MC method is very robust because this convergence rate is independent of the co-dimension of the problem, so its useful to get approximate estimates of very high-dimensional integrals.
Sampling methods with improved convergence rates have been developed, such as Latin Hypercube sampling and Quasi-MC methods. However, the improved convergence rate comes at the expense of several drawbacks, e.g., the convergence of Quasi-MC methods is dependent of the co-dimension of the problem and Latin Hypercube cannot be used for incremental sampling.

Cubature rules. The integrals in (4) can also be computed using cubature rules. These rules are based on a polynomial approximation of the target function, i.e. the function describing the relation between parameters and QoI, so they have super-linear convergence rate on the set of smooth functions. Their applicability is however limited to low-co-dimensional problems because cubature rules based on a tensor grid suffer the *curse of dimensionality*, i.e. if m is the number of points used in the one dimensional rule and d the dimension of the integral, the number of d points at which to evaluate the function grows as $\mathscr{O}(m^d)$. They will however be presented here because they represent a fundamental tool for the creation of high-dimensional model representations that will be presented in the next section.
Let $\mathbf{Z}$ be a vector of *independent* random variables (i.e. $\mathbf{Z}:\Omega\rightarrow R^d$) in the probability space $(\Omega,\mathscr{F},\mu_{\mathbf{Z}})$, where $\mathscr{F}$ is the Borel set constructed on Ω and $\mu_{\mathbf{Z}}$ is the measure associated to $\mathbf{Z}$. By the independence of $\mathbf{Z}$, we can write Ω as a product space $\Omega=\times_{i=1}^{d}\Omega_i$, with product measure $\mu_{\mathbf{Z}}=\times_{i=1}^{d}\mu_i$. For $A\subseteq R^d$, we call $F_{\mathbf{Z}}(A)=\mu_{\mathbf{Z}}(\mathbf{Z}^{-1}(A))$ the distribution of $\mathbf{Z}$.
For each independent dimension of Ω we can construct orthogonal polynomials $\{\phi_n(z_i)\}_{n=1}^{N_i}$, $i=1,\ldots,d$, with respect to the probability distribution F_i, where $F_{\mathbf{Z}}=\times_{i=1}^{d}F_i$ [8]. The tensor product of such basis forms a basis for

$$L^2_{F_{\mathbf{Z}}}=\left\{f:I\subseteq R^d\rightarrow R\,\middle|\,\int_I f^2(\mathbf{z})dF_{\mathbf{Z}}(\mathbf{z})=\mathbf{Var}[f(\mathbf{Z})]<\infty\right\} \tag{6}$$

that means that there exists a projection operator $P_N:L^2_{F_{\mathbf{Z}}}\rightarrow P^N$ such that for any $f\in L^2_{F_{\mathbf{Z}}}$, and with the notation $\mathbf{i}=(i_1,\ldots,i_d)\in[0,\ldots,N_1]\times\ldots\times[0,\ldots,N_d]$,

$$f\approx P_N f=\sum_{\mathbf{i}=0}^{N_1,\ldots,N_d}\hat{f}_{\mathbf{i}}\Phi_{\mathbf{i}},\qquad \hat{f}_{\mathbf{i}}=\frac{(f,\Phi_{\mathbf{i}})_{L^2_{F_{\mathbf{Z}}}}}{\|\Phi_{\mathbf{i}}\|^2_{L^2_{F_{\mathbf{Z}}}}}, \tag{7}$$

where $\Phi_{\mathbf{i}}=\prod_{k\in\mathbf{i}}\phi_k$, $\|f\|^2_{L^2_{F_{\mathbf{Z}}}}=(f,f)_{L^2_{F_{\mathbf{Z}}}}$ and

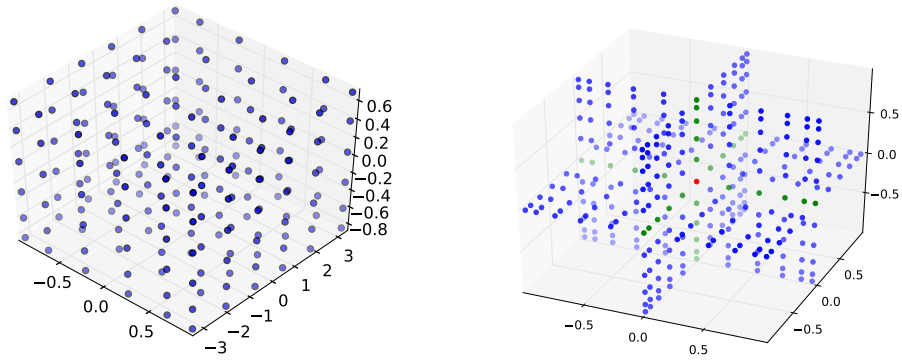
$$(f,g)_{L^2_{F_{\mathbf{Z}}}}=\int_{R^d}f(\mathbf{z})g(\mathbf{z})dF_{\mathbf{Z}}(\mathbf{z}) \tag{8}$$

In the following we will be marginally interested in the approximation (7) of the QoI function. However the fast – possibly spectral – convergence of such approximation is inherently connected with the convergence in the approximation of statistical moments, because $\mu_f=\hat{f}_{\mathbf{0}}$ and $\sigma^2_f=\sum_{\mathbf{i}}\hat{f}^2_{\mathbf{i}}-\hat{f}^2_{\mathbf{0}}$ [9].
From the orthogonal polynomials used in the construction of (7), the 1-dimensional Gauss quadrature points and weights $\left\{z_{j_i},w_{j_i}\right\}_{j_i}^{N_i}$ can be derived using the Golub-Welsch algorithm [8]. Gauss quadrature points and weights $\left\{\mathbf{z}_{j_1,\ldots,j_d},w_{j_1,\ldots,j_d}\right\}_{j_1,\ldots,j_d=1}^{N_1,\ldots,N_d}$ for the tensor product space can be obtained as tensor product of one dimensional cubature rules (see fig. 3a), obtaining the following approximations for (4):

$$\begin{aligned}\mu_{\mathbf{u}}(t) &\approx \bar{\mu}_{\mathbf{u}}(t)=\sum_{j_1}^{N_1}\cdots\sum_{j_d}^{N_d}\mathbf{u}\left(t,\mathbf{z}_{j_1,\ldots,j_d}\right)w_{j_1,\ldots,j_d}\\ \sigma^2_{\mathbf{u}}(t) &\approx \bar{\sigma}^2_{\mathbf{u}}(t)=\sum_{j_1}^{N_1}\cdots\sum_{j_d}^{N_d}\left(\mathbf{u}\left(t,\mathbf{z}_{j_1,\ldots,j_d}\right)-\bar{\mu}_{\mathbf{u}}(t)\right)^2w_{j_1,\ldots,j_d}\end{aligned} \tag{9}$$

Gauss quadrature rules of order N are accurate for polynomials of order up to degree $2N-1$. This high accuracy comes at the expense of the curse of dimensionality due to the use of tensor products in high-dimensional integration. This effect can be alleviated by the use of Sparse Grid technique proposed by Smolyak [10] that uses an incomplete but accurate version of the tensor product. However, in the following section we will see that we can often avoid working in very high-dimensional spaces.

Copyright © 2014 by ASME



(a) Tensor grid (b) cut-HDMR grid

FIGURE 3: EXAMPLE OF THE DISTRIBUTION OF THE POINTS IN TENSOR CUBATURE RULES AND cut-HDMR ACCOUNTING FOR 2nd ORDER INTERACTIONS.

High-Dimensional Model Representation (HDMR)

High-dimensional models are very common in practical applications, where a number of parameters influence the dynamical behavior of a system. These models are very difficult to handle, in particular if we consider them as black-boxes where we are only allowed to change parameters. One method to circumvent these difficulties is the HDMR expansion [11], where the high-dimensional function $f : \Omega \to R$, $\Omega \subseteq R^d$ is represented by a function decomposed with lower order interactions:

$$f(\mathbf{x}) \equiv f_0 + \sum_i f_i(\mathbf{x}_i) + \sum_{i<j} f_{i,j}(\mathbf{x}_i, \mathbf{x}_j) + \cdots + f_{1,2,\ldots,d}(\mathbf{x}_1, \ldots, \mathbf{x}_d). \tag{10}$$

This expansion is exact and exists for any integrable and measurable function f, but it is not unique. There is a rich variety of such expansions depending on the projection operator used to construct them. The most used in statistics is the ANOVA-HDMR where the low dimensional functions are defined by

$$\begin{aligned}
f_0^A &\equiv P_0^A f(\mathbf{x}) = \int_\Omega f(\mathbf{x}) d\mu(\mathbf{x}),\\
f_i^A(\mathbf{x}_i) &\equiv P_i^A f(\mathbf{x}) = \int_{\Omega_i} f(\mathbf{x}) \prod_{i\neq j} d\mu_j(\mathbf{x}_j) - P_0^A f(\mathbf{x}),\\
f_{i_1,\ldots,i_l}^A(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l}) &\equiv P_{i_1,\ldots,i_l}^A f(\mathbf{x}) =\\
&\int_{\Omega_{i_1,\ldots,i_l}} f(\mathbf{x}) \prod_{k\notin\{i_1,\ldots,i_l\}} d\mu_k(\mathbf{x}_k) -\\
&\sum_{k_1<\cdots<k_{l-1}\in\{i_1,\ldots,i_l\}} P_{k_1,\ldots,k_{l-1}}^A f(\mathbf{x}) -\\
&\ldots - \sum_{k\in\{i_1,\ldots,i_l\}} P_k^A f(\mathbf{x}) - P_0^A f(\mathbf{x}),
\end{aligned} \tag{11}$$

where $\Omega_{i_1,\ldots,i_l} \subseteq \Omega$ is the hypercube excluding indices $i_1, \ldots, i_l$ and μ is the product measure $\mu(\mathbf{x}) = \prod_{i=1}^d \mu_i(\mathbf{x}_i)$. This expansion can be used to express the total variance of f, by noting that

$$\begin{aligned}
D &\equiv \mathbf{E}\left[(f - f_0)^2\right] = \sum_i D_i + \sum_{i<j} D_{i,j} + \cdots + D_{1,2,\ldots,d},\\
D_{i_1,\ldots,i_l} &= \int_{\Omega^{i_1,\ldots,i_l}} \left(f_{i_1,\ldots,i_l}^A(\mathbf{x}_{i_1})\right)^2 \prod_{k\in\{i_1,\ldots,i_l\}} d\mu_k(\mathbf{x}_k),
\end{aligned} \tag{12}$$

where $\Omega^{i_1,\ldots,i_l} \subseteq \Omega$ is the hypercube including indices $i_1, \ldots, i_l$. However, the high-dimensional integrals in the ANOVA-HDMR expansion are computationally expensive to evaluate.
An alternative expansion is the cut-HDMR, that is built by superposition of hyperplanes passing through the cut center $\mathbf{y} = (y_1, \ldots, y_d)$:

$$\begin{aligned}
f_0^C &\equiv P_0^C f(\mathbf{x}) = f(\mathbf{y}),\\
f_i^C(\mathbf{x}_i) &\equiv P_i^C f(\mathbf{x}) = f^i(\mathbf{x}_i) - P_0^C f(\mathbf{x}),\\
f_{i_1,\ldots,i_l}^C(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l}) &\equiv P_{i_1,\ldots,i_l}^C f(\mathbf{x}) =\\
&f^{i_1,\ldots,i_l}(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l}) -\\
&\sum_{k_1<\cdots<k_{l-1}\in\{i_1,\ldots,i_l\}} P_{k_1,\ldots,k_{l-1}}^C f(\mathbf{x}) -\\
&\ldots - \sum_{k\in\{i_1,\ldots,i_l\}} P_k^C f(\mathbf{x}) - P_0^C f(\mathbf{x}),
\end{aligned} \tag{13}$$

where $f^{i_1,\ldots,i_l}(\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_l})$ is the function $f(\mathbf{x})$ with all the remaining variables set to $\mathbf{y}$. This expansion requires the evaluation of the function f on lines, planes and hyperplanes passing through the cut center (see fig. 3b).

Copyright © 2014 by ASME

If cut-HDMR (13) is a good approximation of f at order L , i.e. considering up to L-terms interactions in (10), such an expansion can be used for the computation of ANOVA-HDMR in place of the original function. This reduces the computational cost dramatically: let d be the number of parameters and s the number of samples taken along each direction (being them MC samples or cubature points), then the cost of constructing cut-HDMR in terms of function evaluations is

$$\sum_{i=0}^{L} \frac{d!}{(d-i)!i!}(s-1)^i \tag{14}$$

Total Sensitivity Indices

The main task of Sensitivity Analysis is to quantify the sensitivity of the output with respect to the input. In particular it is important to know how much of this sensitivity is accountable to a particular parameter. With the focus on global sensitivity analysis, the sensitivity of the system to a particular parameter can be expressed by the variance of the output associated to that particular input.

One approach to this question is to consider each parameter separately and to apply one of the UQ techniques introduced. This approach goes by the name of *one-at-a-time analysis*. This technique is useful to get a first overview of the system. However, this technique lacks an analysis of the interaction between input parameters, which in many cases is important.

A better analysis can be achieved using the method of Sobol [11]. Here single sensitivity measures are given by

$$S_{i_1,\dots,i_l} = \frac{D_{i_1,\dots,i_l}}{D}, \qquad \text{for } 1 \le i_1 < \cdots < i_l \le n, \tag{15}$$

where D and $D_{i_1,\dots,i_l}$ are defined according to (12). These express the amount of total variance that is accountable to a particular combination $i_1,\dots,i_l$ of parameters. The Total Sensitivity Index (TSI) is the total contribution of a particular parameter to the total variance, including interactions with other parameters. It can be expressed by

$$TS(i) = 1 - S_{\neg i}, \tag{16}$$

where $S_{\neg i}$ is the sum of all $S_{i_1,\dots,i_l}$ that do not involve parameter i.

These total sensitivity indices can be approximated using sampling based methods in order to evaluate the integrals involved in (12). Alternatively, [12] suggests to use cut-HDMR and cubature rules in the following manner:

1. Compute the cut-HDMR expansion on cubature nodes for the input distributions (see fig. 3b),
2. Derive the approximated ANOVA-HDMR expansion from the cut-HDMR,
3. Compute the Total Sensitivity Indices from the ANOVA-HDMR.

This approach gives the freedom of selecting the level of accuracy for the HDMR expansion depending on the level of interaction between parameters. The truncation order L of the ANOVA-HDMR can be selected and the accuracy of such expansion can be assessed using the concept of "effective dimension" of the system: for $q \le 1$,the effective dimension of the integrand f is an integer L such that

$$\sum_{0<|t|\le L} D_t \ge qD, \tag{17}$$

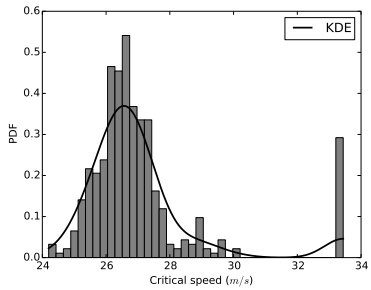
where t is a multi-index $i_1,\dots,i_l$ and $|t|$ is the cardinality of such multi-index. The parameter q is chosen based on a compromise between accuracy and computational cost.

SENSITIVITY ANALYSIS ON RAILWAY VEHICLE DYNAMICS

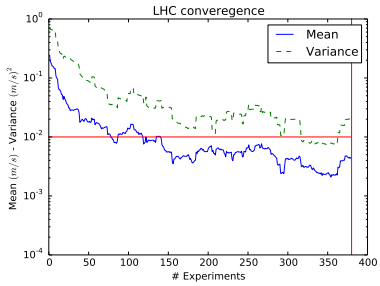
The sensitivity analysis of a dynamical system with respect to its parameters is a computationally expensive task and this cost increases dramatically with the number of parameters. We will adopt the collocation approach presented earlier, thus we will need to obtain an ensemble of solutions. This ensemble is formed by the solutions to the Initial Value Problem IVP (3) for different realizations of the parameters. Each solution is computed using the program DYnamics Train SImulation (DYTSI) developed in [1] with the Explicit Runge-Kutta-Fehlberg method ERKF34 [13]. An explicit solver has been used in light of the analysis performed in [3], where it was found that the hunting motion could be missed by implicit solvers, used with relaxed tolerances, due to numerical damping. In particular implicit solvers are frequently used for stiff problems, like the one treated here, because their step-size is bounded by accuracy constraints instead of stability. However, the detection of the hunting motion requires the selection of strict tolerances, reducing the allowable step-sizes and making the implicit methods more expensive than the explicit ones. Since the collocation approach for UQ involves the computation of completely independent realizations, this allows for a straightforward parallelization of the computations on clusters. Thus, 25 nodes of the DTU cluster have been used to speed up the following analysis.

The first step in the analysis of a stochastic system is the characterization of the probability distribution of the QoI. Since the complete model has co-dimension 48, a traditional sampling method is the best suited for the task of approximating the integrals in eq. (4). In order to speed up the convergence, we used samples generated with the Latin Hyper Cube method [14]. Fig.

Copyright © 2014 by ASME



(a) Histogram of the critical speed.



(b) Convergence of the Latin Hyper Cube

FIGURE 4: APPLICATION OF THE LATIN HYPER CUBE TO OBTAIN THE TOTAL VARIANCE.

4a shows the histogram of the computed critical speeds with respect to the uncertainty in the suspension components. We can notice a big clustering of outliers around $v \approx 33m/s$. This is an indicator of a discontinuity in the parameter space. In particular for such combinations of suspension parameters, the vehicle recovers its stability soon after starting ramping the speed, indicating the merger of the sub-critical Hopf-bifurcation with the fold to a super-critical Hopf-bifurcation. The convergence of the method was checked using a the magnitude of change in the first two estimated moments as shown in figure 4b. Kernel smoothing [7] has been used to estimate the density function according to this histogram. The estimated mean and variance are $\bar{\mu}_v = 27.12m/s$ and $\bar{\sigma}_v^2 = 3.77m^2/s^2$.

One-at-a-time analysis

When each suspension component is considered independently from the others, the estimation problem in (4) is reduced to the calculation of a 1-dimensional integral. This task can be readily achieved by quadrature rules that have proven to be computationally more efficient on problems of this dimensionality than sampling methods [15]. Fourth order quadrature rules have been used to approximate the variances due to the single components. For the 48 parameters describing the suspensions, this leads to the solution of $48 \times 4 + 1 = 193$ Initial Value Problems. The convergence of this method enables a check of accuracy through the decay of the expansion coefficients of the target function [15].
Figure 5a shows the contribution that each suspension component gives to the total variance of the model output. The nomenclature of the components is partly explained in the nomenclature section at the beginning of the paper: for example, `PSTT_RIGHT_K2` stands for the right suspension `K2` (see fig. 1) in the primary suspension connecting the trailing wheel set to the trailing bogie frame. We notice that the analysis doesn't explain the whole variance, but only half of it. This means that some of the variance must be explained by the combined contribution by several parameters.
This first analysis is anyway useful to get a first selection of the most relevant suspensions in the system. Table 2 shows the value of the variance due to the most relevant components. The remaining components contribute less than $0.02m^2/s^2$ each.

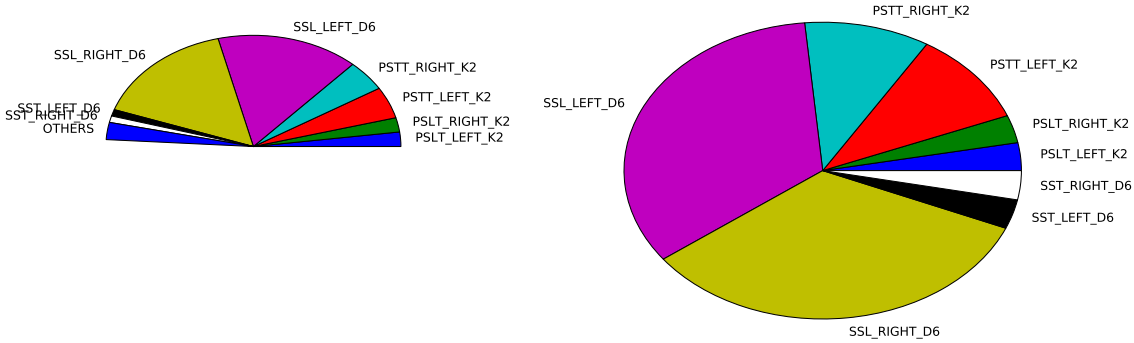
Suspension	One-at-time	ANOVA
	σ_v^2	TSI
`PSLT_LEFT_K2`	0.08	0.09
`PSLT_RIGHT_K2`	0.08	0.09
`PSTT_LEFT_K2`	0.17	0.24
`PSTT_RIGHT_K2`	0.17	0.24
`SSL_LEFT_D6`	0.59	1.07
`SSL_RIGHT_D6`	0.59	1.07
`SST_LEFT_D6`	0.04	0.15
`SST_RIGHT_D6`	0.04	0.15

TABLE 2: SENSITIVITIES OF THE MOST RELEVANT SUSPENSIONS.

Total Sensitivity Analysis

The calculation of the total sensitivity analysis through the use of the cut-HDMR representation and of high order quadrature rules, allows to take into account the interaction between parameters and at the same time limits the amount of computations required exploiting the fast convergence of the quadrature rules.

Copyright © 2014 by ASME



(a) One-at-a-time analysis.

(b) ANOVA analysis

FIGURE 5: APPLICATION OF THE ONE-AT-TIME AND ANOVA ANALYSIS.

The complete sensitivity problem involves 48 parameters. For the full sensitivity analysis using cut-HDMR truncated at second order interaction and with second order quadrature rules, this would result in $1+2\times 48+4\times\binom{48}{2}=4609$ solutions of the deterministic problem. Even if this is affordable in approximately 4 days using 25 nodes of the DTU cluster, we decided to use the fact that the One-at-a-time analysis already provided a good indication of which components would be the most relevant. We use this information to perform a more accurate Total Sensitivity Analysis on the eight suspension components identified before. The remaining suspension coefficients are set to their nominal values. Of course this refinement is susceptible to errors if the underlying function is particularly pathological.

The cut-HDMR representation is truncated at second order interactions, with fourth order quadrature rules. The construction of such surrogate requires the computation of $1+4\times 8+16\times\binom{8}{2}=481$ solutions to the deterministic problem (2). Figure 5b and table 2 show the Total Sensitivity Indices for the suspension parameters. The total variance represented by this analysis is sufficient to explain all the variance of the model output, indicating that the effective dimensionality – see (17) – of the model is $L=2$. Actually, the total variance computed using the cut-HDMR representation exceeds the total variance computed using the Latin Hyper Cube method. This is due to both the additional computational noise introduced by the heuristic for the detection of the critical speed and a discontinuity in the parameter space that makes the sub-critical Hopf-bifurcation merge with the fold to create a super-critical Hopf-bifurcation, as shown in the histogram in figure 4a.

Discussion of the obtained results

The sensitivity analysis of the rail car, running at hunting speed on a track with a curve radius of $1600m$ and super-elevation of $110mm$, reveals that the key parameters determining the critical speed are the yaw dampers in the secondary suspensions and the yaw springs in the trailing primary suspensions in the leading and trailing bogie frames. These components are expected to have an important role in the steering of the car in the curve and the yaw dampers were historically introduced to stabilize the dynamics of rail cars.

However, we remind the reader that these results are strongly conditioned by the choice of the distributions describing the suspension parameters. If different distributions are used, maybe based on the observation of the manufacturing uncertainty of the suspension coefficients, the results could change drastically.

Remarks on uncertainty quantification and sensitivity analysis

The first question that an engineer performing analysis of a stochastic model has to wonder about is whether the uncertain input parameters considered are independent from a probabilistic point of view (we remind that the events A,B are independent if $P(A\cap B)=P(A)P(B)$) or at least uncorrelated. In motivating our example of the uncertainty on the suspension components, we mentioned that their values are uncertain at the manufacturing time. This uncertainty is even more relevant after thousands of running kilometers, due to the wear. However the two cases are slightly different: in the first case the value of each component can be considered independent and uncorrelated from the others, instead in the second case the wear on each of the components cannot be considered independent from the others, because they

Copyright © 2014 by ASME

undergo coupling dynamics! A variety of techniques exist to deal with this problem, in order to find a map from the high dimensional correlated random variables to a lower dimensional set of uncorrelated ones. This however goes beyond the scope of this work. We refer the reader to [9] for a short introduction to the problem.

CONCLUSIONS

Sensitivity analysis is of critical importance in a wide range of engineering applications. The traditional approach of local sensitivity analysis is useful in order to characterize the behavior of a dynamical system in the vicinity of the nominal values of its parameters, but it fails in describing wider ranges of variations. The global sensitivity analysis aims at representing these bigger variations and at the same time it embeds the probability distributions of the parameters in the analysis. This enables the engineer to take decisions, such as improving a design, based on the partial knowledge of the system.

Wrongly approached, a global sensitivity analysis can turn to be a computationally expensive or even prohibitive task. In this work a collection of techniques are used in order to accelerate such analysis for a high-co-dimensional problem. Each of the techniques used allows for a control of the accuracy, e.g., in terms of convergence rate for the cubature rules and the "effective dimension" of the model. This makes the framework flexible and easily adaptable to problems with more diversified distributions and target functions.

The analysis performed on the complete car running in the curve with radius $1600m$ and super-elevation $110mm$ showed that the steering suspension components account for most of the variance of the system, meaning that their coefficient values must be carefully monitored.

It is important to notice that the same settings for global sensitivity analysis can be used for the investigation of different Quantities of Interests, such as wear in curved tracks, angle of attack etc., once they have been properly defined. Furthermore, the "non-intrusive" approach taken allows the engineer to use closed software for the computations. The machinery for sensitivity analysis needs only to be wrapped around it, without additional implementation efforts.

REFERENCES

[1] Bigoni, D., 2011. "Curving Dynamics in High Speed Trains". Master's thesis, The Danish Technical University.

[2] True, H., 1999. "On the theory of nonlinear dynamics and its applications in vehicle systems dynamics". *Vehicle System Dynamics,* ***31***(5-6), pp. 393–421.

[3] True, H., Engsig-Karup, A. P., and Bigoni, D., 2014. "On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics". *Mathematics and Computers in Simulation,* ***95***, pp. 78–97.

[4] Kik, W., and Moelle, D., 2010. ACRadSchiene - To create or Approximate Wheel/Rail profiles - Tutorial. Tech. rep.

[5] Kalker, J., 1991. "Wheel-rail rolling contact theory". *Wear,* ***144***(1-2), Apr., pp. 243–261.

[6] Shen, Z., Hedrick, J., and Elkins, J., 1984. "A comparison of alternative creep-force models for rail vehicle dynamic analysis". In 8th IAVSD Symposium.

[7] Hastie, T., Tibshirani, R., and Friedman, J., 2001. *The elements of statistical learning*, Vol. 1. Springer Series in Statistics.

[8] Gautschi, W., 2004. *Orthogonal Polynomials: Computation and Approximation*. Numerical Mathematics and Scientific Computation. Oxford University Press.

[9] Xiu, D., 2010. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, July.

[10] Petras, K., 2003. "Smolyak cubature of given polynomial degree with few nodes for increasing dimension". *Numerische Mathematik,* ***93***(4), Feb., pp. 729–753.

[11] Saltelli, A., Chan, K., and Scott, E. M., 2008. *Sensitivity Analysis*. John Wiley & Sons, Dec.

[12] Gao, Z., and Hesthaven, J., 2011. "Efficient solution of ordinary differential equations with high-dimensional parametrized uncertainty". *Communications in Computational Physics,* ***10***(2), pp. 253–286.

[13] Hairer, E., and Wanner, G., 1991. *Solving ordinary differential equations II: Stiff and differential-algebraic problems*, Vol. 14 of *Springer series in computational mathematics*. Springer-Verlag.

[14] Mckay, M., Beckman, R., and Conover, W., 2000. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code". *Technometrics,* ***41***(1), pp. 55–61.

[15] Bigoni, D., Engsig-Karup, A., and True, H., 2012. "Comparison of Classical and Modern Uncertainty Quantification Methods for the Calculation of Critical Speeds in Railway Vehicle Dynamics". In 13th mini Conference on Vehicle System Dynamics, Identification and Anomalies.

Copyright © 2014 by ASME

Available online at www.sciencedirect.com
SciVerse ScienceDirect
Mathematics and Computers in Simulation xxx (2012) xxx–xxx
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original article

On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics

H. True*, A.P. Engsig-Karup, D. Bigoni

DTU Informatics, The Technical University of Denmark, Asmussens Alle 305, DK-2800 Kgs Lyngby, Denmark

Received 1 September 2011; received in revised form 8 August 2012; accepted 28 September 2012

Abstract

The paper contains a report of the experiences with numerical analyses of railway vehicle dynamical systems, which all are nonlinear, non-smooth and stiff high-dimensional systems. Some results are shown, but the emphasis is on the numerical methods of solution and lessons learned. But for two examples the dynamical problems are formulated as systems of ordinary differential-algebraic equations due to the geometric constraints. The non-smoothnesses have been neglected, smoothened or entered into the dynamical systems as switching boundaries with relations, which govern the continuation of the solutions across these boundaries. We compare the resulting solutions that are found with the three different strategies of handling the non-smoothnesses. Several integrators – both explicit and implicit ones – have been tested and their performances are evaluated and compared with respect to accuracy, and computation time.
© 2012 IMACS. Published by Elsevier B.V. All rights reserved.

Keywords: Railway vehicle dynamics; Stiff systems; Nonlinear dynamics; Non-smoothness

1. Introduction

In recent years the world has seen a rapid development of theoretical research in the area of non-smooth dynamical systems. This development is a natural extension of the mathematical theory of nonlinear dynamical systems that are assumed 'sufficiently smooth', which usually means that all partial derivatives of second order in the dynamical system must be continuous. The area of mathematical theory of nonlinear sufficiently smooth dynamical systems grew very fast in the 20th century. In the second half of the century the development was strongly fueled by the growing application of digital computers and efficient numerical methods that together made it possible to solve nonlinear dynamical problems that hitherto had not been solvable with the known analytic solution methods.

In real life, however, the dynamical problems often do not satisfy the 'sufficiently smooth' criterion, and many of the mathematical results are not valid any more. Solutions of non-smooth problems were therefore limited in number but of important examples the theory of the clock and the motion of a body under the action of a Coulomb type friction force ought to be mentioned. These dynamical systems were simple one degree of freedom systems. As examples of the breakdown of the mathematical theory for smooth dynamical systems we mention the center manifold theorem for bifurcations and the necessary condition for existence of a bifurcation, which both do not hold in general for

* Corresponding author. Tel.: +45 4525 3016.
E-mail address: ht@imm.dtu.dk (H. True).

0378-4754/$36.00 © 2012 IMACS. Published by Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.matcom.2012.09.016

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

non-smooth dynamical systems. Instead numerical continuation routines must be applied to find bifurcation points and multiple attractors. This fact emphasizes the importance of accurate and reliable numerical methods for the analysis of theoretical dynamical problems.

Nonlinear dynamical models of mechanical systems with more than two degrees of freedom (DOF) are in general not solvable by analytic methods, but with the development of the digital computer the impossible became possible, and a vast amount of dynamical models of mechanical systems of interest for the applications could then be analyzed. Vehicle system dynamics is one of such mechanical systems.

Complete dynamical models of vehicle systems are nonlinear and non-smooth with degrees of freedom (DOFs) in the range from about 10 to above 100. It is therefore necessary to find the solutions by numerical methods. In the beginning of the age of numerical investigations of vehicle system dynamics the numerical routines were rather crude, mostly explicit formulations with fixed step length and error tolerance. No special attention was given to the non-smoothnesses in the problem, but they often caused problems of their own. The interest in the numerical methods was limited to the question: Do I get an answer? If 'yes', then fine. An investigation of numerical integration methods for vehicle dynamical problems is found in chapter 2 in the book by Garg and Dukkipatti [7] from 1984. It describes the state of the art at that time. The authors mainly compare explicit and implicit solution routines and show the relative performances of several integration schemes from that time. No attention is paid to the handling of non-smoothnesses in the system.

Around the same time a production of simulation routines for modeling and analysis of vehicle dynamical systems started around the world. New integration routines were developed that were especially designed for vehicle dynamical use. Characteristic for vehicle dynamical systems is the mathematical formulation as a differential-algebraic dynamical problem, which is very stiff. The routine DASSL deserves to be mentioned in this context. Several of these routines are commercially available and have been further developed and used successfully in both industry and research institutes. Some of them participated in a Manchester benchmark test [15] in 1998, where their performances were compared.

In this article we will describe the development of the numerical handling of non-smooth vehicle dynamical systems at The Technical University of Denmark. On the background of some results with bifurcations of as well periodic, quasi-periodic as chaotic solutions and the existence of multiple attractors, we discuss the use of various numerical solution routines. In the long period of applications we have investigated problems with discontinuous second derivatives, discontinuous first derivatives and discontinuous functions. The size of the dynamical problems varies from low-dimensional test problems to high-dimensional realistic railway vehicle models. We have solved these problems as well by ignoring the non-smoothnesses as by smoothing them and by introduction of switching boundaries with event detection. We have compared the solutions that resulted from the various approaches and tried to select a solution strategy that would perform in an optimal way for each given problem. We would like to share our experiences with other members of the scientific and industrial community. In a final section of the article we shall compare the performance of several of the routines we have used and give recommendations for their applications to non-smooth dynamical problems on the basis of our experience.

The reader, who wants information about the general problem of nonlinear railway dynamics, may find [34] a useful reference. An article by Knothe and Böhm [21] describes the history of railway dynamics and both contributions [34,21] contain many references for further studies. The EUROMECH 500 workshop entitled 'Non-smooth Problems in Vehicle Systems Dynamics' was held in 2008, and the contributions are published in the proceedings [32].

A very informative state-of-the art article on numerical methods in vehicle system dynamics by Arnold et al. [1] has recently been published. It is a valuable evaluation and a description of the use of the various available integration routines for vehicle system dynamical problems that exist today.

2. Theoretical basis for railway vehicle dynamical systems

The theoretical model of the dynamics of railway vehicles is usually formulated as a dynamical multibody system under external forcing. The single bodies are most often assumed rigid. Flexible bodies may appear, but the flexibilities are then often represented by a Galerkin approximation of their characteristic frequencies of deformation in order to avoid the modeling of the dynamics of the flexible bodies by partial differential equations. The internal forces between the bodies can be classified in two main groups: (i) spring and damper forces and (ii) contact forces. The spring and damper forces are in general nonlinear, and the contact forces, which always are nonlinear, can be divided into rolling

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

contact, sliding contact with stick/slip and impacts. All these forces introduce non-smoothnesses in the dynamical system.

The dynamical system depends on several parameters from which the speed, V, is usually chosen as the control parameter in a co-dimension 1 problem. In some applications other control parameters may appear, e.g., in curving, where the radius of the curve and the so-called super elevation, which describes the slope of a cross-section of the track, are important independent parameters. All other parameters are considered constant.

If N is the number of degrees of freedom of the multibody problem with time, t, as the independent variable and $\mathbf{P}$ a set of independent parameters, then we obtain the $2N$ state variables $x_i(t;\mathbf{P})$, $1 \le i \le 2N$, and the dynamical system can be written as a general nonlinear initial value problem on the form

$$\frac{d\mathbf{x}}{dt} = \mathbf{F}(\mathbf{x}, t; \mathbf{P}), \quad t > 0 \tag{1}$$

with appropriate initial conditions $\mathbf{x}(0)$ where M is the number of independent parameters. Thus, $\mathbf{x}(t) \in \mathbb{R}^{2N}$ and $\mathbf{P} \in \mathbf{R}^M$. The vector function $\mathbf{F}(\mathbf{x}, t; \mathbf{P}) \in \mathbb{R}^{2N}$ is a nonlinear and in general a non-smooth function of its arguments.

In addition there are constraint equations. For each wheel set of K total sets an equation of the form

$$\frac{dx_k}{dt} = f_k(\mathbf{x}, t; \mathbf{P}), \quad k = 1, 2, \ldots, K \tag{2}$$

expresses that the two wheels on the axle rotate with the same angular velocity (the axle is assumed rigid) or with different velocities when an elastic connection between the wheels is assumed. The rolling contact parameters of the wheel/rail contact surface are calculated on the basis of the geometrical contours of the two bodies, their relative orientations and the normal load in the contact surface. In real life the relations are non-smooth, and must be evaluated numerically and tabulated. The condition that the wheels and rails are in contact is expressed by a set of constraint equations that combine the kinematic contact variables in a nonlinear relation. These relations together with other possible contact conditions between the bodies in the system constitute a set of constraint equations. These reduce the number of generalized coordinates in the problem to a value below N. Under the influence of dynamical forces on the system some of these relations may become time dependent. The sudden changes in the number of generalized coordinates – for example if a wheel lifts off from the rail – introduces additional non-smoothnesses in the system.

The wheel/rail forces in the rolling contact are explicitly formulated as nonlinear relations between the normal and tangent forces in the contact surface on one side and the deformation under normal load and the normalized accumulated tangential strain velocities in the contact surface – the so-called creepage – on the other. The resulting tangent forces – denoted the creep forces – depend non-linearly on the normal forces, the wheel/rail contact geometry and the creepage. Since the contact surface kinematic relations depend non-smoothly on the relative orientations of the wheel and the rail so do the wheel/rail forces. All non-smoothnesses in the dynamical problem including those that represent sliding contact and impacts should be defined by the switching boundaries $h_j(\mathbf{x}) = 0$, where $1 \le j \le J$, and J is the number of non-smoothnesses with corresponding relations. More about that in Section 5.2.

In this article we mainly consider equilibrium solutions of the dynamical problems, therefore the dynamical systems become autonomous.

The dynamics of a complete wagon model running on straight track or in canted curved tracks can be studied using the Newton–Euler formulation of the dynamical system. Several reference frames are introduced in order to simplify the description of the system:

- the *inertial reference frame I* cannot be used because the model is quickly moving and the dynamics that need to be observed are in the order of the 10^{-3}–10^{-6} m.
- a *track following reference frame F* is attached to the centerline of the track, at the level of the height of the rails, and it moves with the train. This reference frame can be inertial if the track is straight and the train moves at constant speed. Otherwise the frame is not inertial and fictitious forces need to be added to the system.
- each body has its own reference frame, called the *body following reference frame* that is attached to the center of mass of the body.
- additional reference frames, called the *contact point reference frames*, can be used for the modeling of wheel–rail contact forces.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

For each body in the system the Newton–Euler relations hold:

$$\sum_{i=1}^{n} {}^{I}\vec{F}_i = m\vec{a} \quad (\text{Newton}'\text{s } law) \tag{3}$$

$$\sum_{i=1}^{m} {}^{B}\vec{M}_i = \frac{d}{dt}({}^{B}[J]{}^{B}\vec{\omega}) + {}^{B}\vec{\omega} \times ({}^{B}[J]{}^{B}\vec{\omega}) \quad (\text{Euler}'\text{s law}) \tag{4}$$

where ${}^{I}\vec{F}_i$ and ${}^{B}M_i$ are, respectively, the forces and torques acting on the center of mass, m and $[J]$ are the mass and the tensor moment of inertia respectively, $\vec{a}$ and $\dot{\vec{\omega}}$ are the linear acceleration and the angular acceleration of the bodies. The left superscripts stand for the inertial or the body reference frame. Fictitious forces and torques are added in order to be able to write all the equations of motion in the track following reference frame. The simplification of negligible torque terms leads to the following fictitious forces:

$${}^{F}\vec{F}_c = \begin{Bmatrix} 0 \\ m\left[\dfrac{v^2}{R}\cos(\phi_t)\right] \\ m\left[-\dfrac{v^2}{R}\sin(\phi_t)\right] \end{Bmatrix} \tag{5}$$

where v is the speed of the train, R is the radius and ϕ_t is the cant of the track in the curve and the superscript F indicates that the force is written in the track following reference frame. All the bodies will be subject to the gravitational forces as well:

$${}^{F}\vec{F}_g = \begin{Bmatrix} 0 \\ -mg\sin\phi_t \\ -mg\cos\phi_t \end{Bmatrix} \tag{6}$$

The contact forces can be split in guidance forces, determined by the normal load and the positive conicity of the wheels, and creep forces, due to the shear and sliding of the wheels on the rails. For the modeling of these forces, several approximations exist, that go from the use of a stiff non-linear spring to the realistic approach to the contact problem. The notation ${}^{F}\vec{F}_{N_l}$ and ${}^{F}\vec{F}_{N_r}$ will be used to refer to the guidance forces on the left and the right wheel of a wheel set. In the same way, the notation ${}^{F}\vec{F}_{C_l}$ and ${}^{F}\vec{F}_{C_r}$ will be used for the creep forces. The total forces on the left and the right wheel of a wheel set will be denoted by ${}^{F}\vec{F}_L$ and ${}^{F}\vec{F}_R$ respectively. The torques due to these forces will also be considered and will be denoted by ${}^{B}\vec{M}_L$ and ${}^{B}\vec{M}_R$.

The last groups of forces that are applied to all the bodies are the suspension forces. Each element of the suspension, generically called *link*, will be characterized by a function f such that

$$\begin{Bmatrix} {}^{F}\vec{F}_l \\ {}^{F}\vec{T}_l \end{Bmatrix} = f({}^{F}\vec{b}_{l_0}, {}^{F}\vec{b}_l, {}^{F}\vec{v}_l, {}^{F}\vec{\theta}_l, {}^{F}\dot{\vec{\theta}}_l) \tag{7}$$

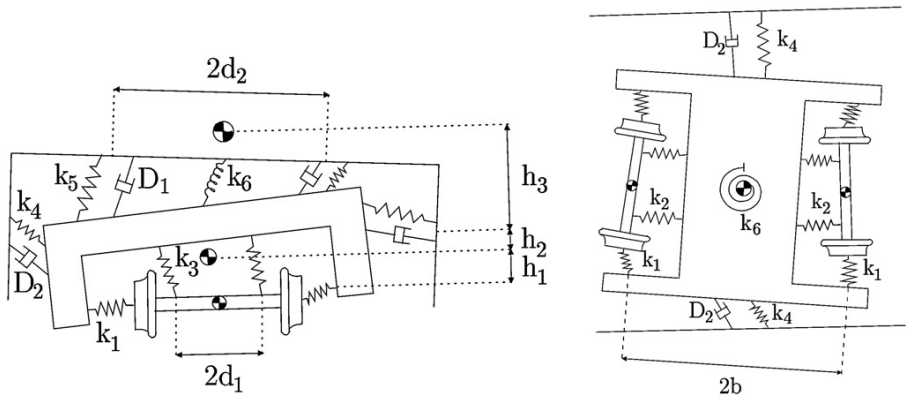
where ${}^{F}\vec{b}_{l_0}$ is the length of the link at rest, ${}^{F}\vec{b}_l$ is the deformed length of the link, ${}^{F}\vec{v}_l$ is the relative speed of the two attack points of the link, ${}^{F}\vec{\theta}_l$ is the deformed angle between the bodies connected by the link and ${}^{F}\dot{\vec{\theta}}_l$ is the angular velocity of the bodies. These quantities can be easily computed using basic geometry, knowing the positions at which the links are connected and the state of the dynamics. The characteristic function of the link, that is usually non-linear, will determine the resulting forces and the torques. Each suspension system is a collection of spring and damping elements. The total resulting forces and torques due to the ith suspension system will be denoted by ${}^{F}\vec{F}_s^i$ and ${}^{B}\vec{M}_s^i$.

Substituting the gravitational, the centrifugal and the suspension forces in (3) and (4), the equation of motion (EOM) of the car body can be obtained.

$$m\ddot{\vec{x}} = {}^{F}\vec{F}_g^C + {}^{F}\vec{F}_c^C + {}^{F}\vec{F}_s^{SS_l} + {}^{F}\vec{F}_s^{SS_t} \tag{8}$$

$$[J]\dot{\vec{\omega}} = {}^{B}\vec{M}_g^C + {}^{B}\vec{M}_c^C + {}^{B}\vec{M}_s^{SS_l} + {}^{B}\vec{M}_s^{SS_t} \tag{9}$$

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016



(a) Front view of the bogie (b) Top view of the bogie

Fig. 1. The Cooperrider bogie model.

where the superscript C stands for the car body and $SS_{l/t}$ indicates the leading and trailing secondary suspensions. Similarly, the EOM of the leading bogie frame can be obtained:

$$m\ddot{\vec{x}} = {}^{F}\vec{F}_g^{B_l} + {}^{F}\vec{F}_c^{B_l} + {}^{F}\vec{F}_s^{SS_l} + {}^{F}\vec{F}_s^{PS_{ll}} + {}^{F}\vec{F}_s^{PS_{lt}} \tag{10}$$

$$[J]\dot{\vec{\omega}} = {}^{B}\vec{M}_g^{B_l} + {}^{B}\vec{M}_c^{B_l} + {}^{B}\vec{M}_s^{SS_l} + {}^{B}\vec{M}_s^{PS_{ll}} + {}^{B}\vec{M}_s^{PS_{lt}} \tag{11}$$

where B stands for the leading bogie frame and $PS_{ll/lt}$ indicate, respectively, the leading and trailing primary suspensions. A similar notation is used for the trailing bogie frame. Since the wheel sets are spinning on the track, the pitch angle is not relevant. However, the angular velocity is important in the computation of the creepages as it is given by the nominal spinning speed $\frac{v}{r_0}$ and the speed perturbation β due to the odd distribution of the forces among the wheels. The resulting equations of motion for the leading wheel set attached to the leading bogie frame can be written as:

$$m\ddot{\vec{x}} = {}^{F}\vec{F}_g^{W_{ll}} + {}^{F}\vec{F}_c^{W_{ll}} + {}^{F}\vec{F}_L^{W_{ll}} + {}^{F}\vec{F}_R^{W_{ll}} + {}^{F}\vec{F}_s^{PS_{ll}} \tag{12}$$

$$\begin{aligned} J_\phi\ddot{\phi} = {} & \left\{{}^{B}M_L^{W_{ll}}\right\}_\phi + \left\{{}^{B}M_R^{W_{ll}}\right\}_\phi \\ & + \left\{{}^{B}\vec{M}_g^{W_{ll}}\right\}_\phi + \left\{{}^{B}\vec{M}_c^{W_{ll}}\right\}_\phi + \left\{{}^{B}\vec{M}_s^{PS_{ll}}\right\}_\phi \end{aligned} \tag{13}$$

$$J_\chi\dot{\beta} = \left\{{}^{B}M_L^{W_{ll}}\right\}_\chi + \left\{{}^{B}M_R^{W_{ll}}\right\}_\chi \tag{14}$$

$$\begin{aligned} J_\psi\ddot{\psi} = {} & \left\{{}^{B}M_L^{W_{ll}}\right\}_\psi + \left\{{}^{B}M_R^{W_{ll}}\right\}_\psi \\ & + \left\{{}^{B}\vec{M}_g^{W_{ll}}\right\}_\psi + \left\{{}^{B}\vec{M}_c^{W_{ll}}\right\}_\psi + \left\{{}^{B}\vec{M}_s^{PS_{ll}}\right\}_\psi \end{aligned} \tag{15}$$

where W stands for wheel set and the resulting forces are given by the sum of gravitational, centrifugal, suspension and contact forces. The notation $\{\vec{a}\}_i$ stands for the ith component of the vector $\vec{a}$. Similar equations can be derived for the remaining wheel sets of the model.

Depending on the level of accuracy that is wanted, assumptions can be made in order to simplify the model. For example the car body could be considered fixed if only the dynamics of the wheel sets and the bogie frames need to be analyzed.

3. Models with impact

The dynamics of Cooperrider's bogie model [4] has been investigated in detail. The model is shown in Fig. 1. A detailed description of the model is presented by Kaas-Petersen in [17] (notice a printing error on p. 92. G is correctly 8.08×10^{10} N/m^2). The important features of the model are that the vertical motions are assumed to be so small that the

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

coupling with the other degrees of freedom can be neglected, and the dynamical system therefore is reduced to a system of 14 first order differential equations that describe the horizontal motion of the bogie elements. All bodies are rigid, the wheel/rail kinematics and the spring and damper constitutive relations are linearized, so the only nonlinearities in the system are the contact forces between the wheels and the rails. There is a $u|u|$ term in the wheel/rail creepage/creep force relation, where u denotes the creepage. It means that the second derivative of the relation does not exist in $u = 0$. The action of the wheel flange is modeled by a very stiff linear restoring spring with a dead band δ. With q denoting the lateral displacement of a wheel set, this leads to a non-smoothness at $q = \pm\delta$, where a jump in the first derivative occurs.

The trivial solution satisfies the system for all values of the speed V, but it loses stability in a subcritical bifurcation at the speed V_H. The $u|u|$ term in the creepage/creep force relation changes the initial growth of the bifurcating periodic branch from a square root to a linear function (see True [33]), and the restoring spring creates a tangent bifurcation that stabilizes the oscillation at the lower speed the so-called critical speed V_C. At higher speeds of the bogie chaos develops (see Kaas-Petersen [17], Jensen [16] and Isaksen and True [14]).

The problem is solved numerically. Kaas-Petersen's continuation routine PATH [18] is used to calculate the bifurcation diagram for the dynamical system. PATH also calculates the eigenvalues of the Jacobian and estimates the Floquet multipliers of the Poincaré map in order to determine the stability of the various branches. Its most important feature is that it uses a mixture of time integration and Newton iteration to find the periodic solutions, whereby the computational work is reduced. A periodic solution is treated as the identity under a Poincaré map. In this way the program determines the stable and unstable solutions with the same accuracy. The Poincaré section is chosen by PATH in such a way that it is 'sufficiently transversal' to the phase space trajectory. For the numerical integrations the LSODA routine is used, which automatically switches between stiff and non-stiff solution methods whenever needed (see Petzold [26]). PATH determines the solutions with a relative error of 10^{-9}.

In the points $q = \pm\delta$ the Jacobian is not defined, and two possible ways to handle the non-smoothness were tried. First the singularity was smoothed by a hyperbolic cosine function around $q = \pm\delta$ and second the singularity was neglected and the integration simply continued across the singularity. Since no difference in the resulting dynamics could be detected, and the computation time was almost the same, the second way was chosen in the numerical investigation.

Knudsen et al. [22] and Slivsgaard and True [30] investigated the dynamics of a single-axle bogie, which is essentially only one half of the Cooperrider bogie. Knudsen proved the existence of chaos produced by the singularity in $q = \pm\delta$. For the numerical integrations Knudsen used as well the LSODA routine as an eight-stage explicit Runge–Kutta pair of order five and six. It uses variable time step and error control. To approximate the solution between the integration steps an interpolant with an asymptotic error of the same order as the global error for the numerical integration was used. The method was developed by Enright et al. [5]. This solver was chosen because it should be particularly well suited for the shadowing of a chaotic attractor. Knudsen observed that the flange forces changed continuously across the singularity in $q = \pm\delta$ and therefore the singularity was ignored in both integration methods.

Slivsgaard and True [30] also used PATH and found that the bifurcation of the periodic solution from the trivial solution is supercritical, and that the initial growth of the periodic attractor with the speed is linear. When $|q| = \delta$ a grazing bifurcation takes place and the motion becomes chaotic. It is interesting to compare the result with the bifurcations in the Cooperrider bogie model [4]. In the Cooperrider model a tangent bifurcation stabilizes the unstable periodic branch when $|q|$ grows through δ, but in Slivsgaard's single-axle bogie model the stable periodic motion becomes chaotic in a grazing bifurcation.

All the dynamical systems described above did not include the constraint of the rigid axle.

The investigation of a complete wagon model has recently been performed by Bigoni in [2]. The model employed two Cooperrider bogies attached to a car body and four wheel sets with profile S1002. Fig. 2 shows the design of the model and the location of the suspension elements. The original Cooperrider bogie uses torsional springs and dampers in the secondary suspension. They have been substituted by yaw springs and dampers. The suspension elements can be linear or non-linear.

The rail profile UIC60 with cant 1/40 combined with the wheel profile S1002 cause the appearance of multiple contact points for certain displacements of the wheel sets. These are approximated by a single patch using the method proposed by Sauvage and Pascal [28]. The static parameters for the computation of the contact forces have been obtained using the RSGEO [20] routine. The normal load can be found using the Hertz's contact theory [10] and adjusting the value with the additional penetration due to the dynamics using Kalker's work [19]. The creep forces were found using the Shen, Hedrick and Elkins non-linear theory [29].

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

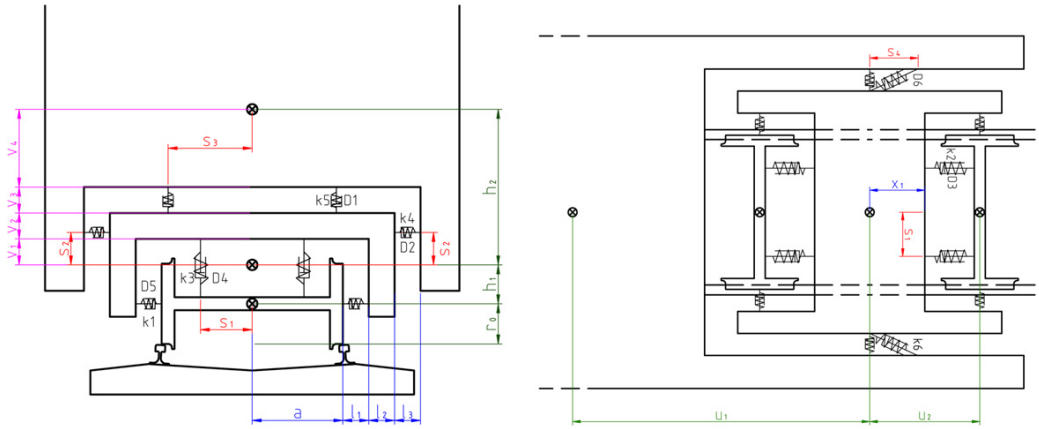


Fig. 2. Design of the Cooperrider bogie attached to a car body.

Using the formulation of the multibody problem introduced in Section 2, a system of 66 coupled first-order differential equations has been obtained. The system can be simplified using superposition when only suspension elements with linear characteristic function are used. Also the computation of the Jacobian can be sped up using the analytical values for the parts that have linear functions and using difference approximation for the wheel sets, where the contact forces are the only non-linear part of the system. These simplifications cannot be performed if the model employs non-linear suspension elements.

The dynamical problem was solved numerically using the Explicit Singly Diagonal Implicit Runge–Kutta (ESDIRK) method with appropriate initial conditions for increasing values of the speed. The ESDIRK method by Nielsen–Thomsen (ESDIRK34 NT1) [24] is a Runge–Kutta method of order 3 for the solution of stiff systems of ODE's and index one DAE's. The type of method is a 4-stage generalized linear method that is reformulated in a special semi-implicit Runge–Kutta method. The error estimation is by imbedding a method of order 4 based on the same stages as the method and the coefficients are selected for ease of the implementation. The method has 4 stages and the stage order is 2. For purposes of generating a dense output and for initializing the iteration in the internal stages a continuous extension is derived. The method is *A*-stable.

4. Models with dry friction contact

In mechanical systems with dry friction contact, with stick/slip between some bodies in the system, the degrees of freedom of the system will vary with the changes of the acting dry friction force vector. Such a system is often referred to as a structure varying system or a structural variant system. In these systems the switching boundaries that were mentioned in Section 2 must be introduced in the state space in order to define the location of the non-smoothnesses. At the switching boundaries the switch conditions must be formulated in order to define the initial conditions for the continuation of the integration of the dynamical system in the appropriate domain of the state space. In this section only one-dimensional dry friction forces occur.

Our first dynamical model of a railway vehicle with dry friction dampers with stick/slip was set up to investigate the interaction between the nonlinear dry friction damping and the nonlinear wheel/rail creep forces. Therefore the model should be so simple that the dynamical features easily can be related to this interaction without interference from other sources. True and Asmund [35] therefore started the analysis with a model of a modification of half the Cooperrider bogie. Fig. 3 illustrates the model. The stiff spring model of the action of the wheel flange in the original Cooperrider bogie was left out, and the linear wheel/rail kinematic relation and the linear characteristic of the spring was kept in place. This of course might result in unrealistically large amplitudes of the lateral motion of the wheel set.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

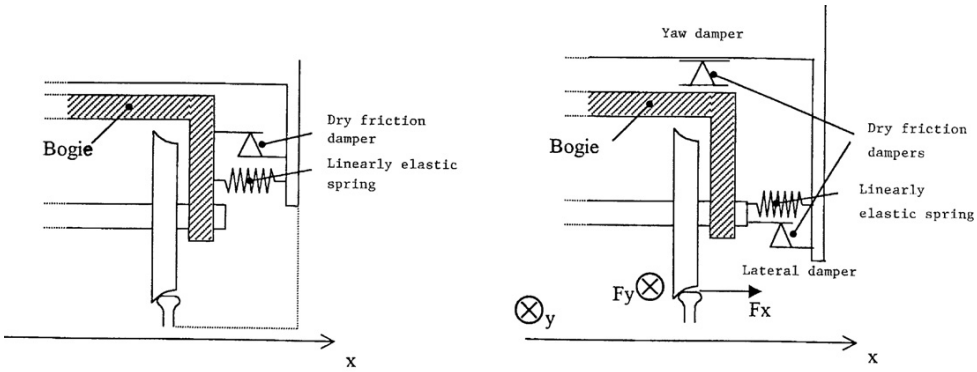


Fig. 3. The single-axle bogie with lateral dry friction damper (left) and with lateral and yaw dry friction damper (right).

The modeling of the stick/slip action in the dry friction is crucial. In order to control the jump from stick to slip in the friction relation a new heuristic smooth transition was developed and tried on some simple test cases. The results were satisfactory, and the dry friction model was therefore adopted for the vehicle model.

First the bifurcation diagram of the model with linear dampers was calculated. The dampers were laid out in such a way that the dissipation in one period of the oscillation would be approximately the same as the dissipation of the dry friction damper. Then the bifurcation diagram for the same model but now with a lateral dry friction damper and no yaw damper was calculated. The two bifurcation diagrams were plotted for comparison in Fig. 4. It is interesting to note that with the dry friction damper the bifurcation disappears and a periodic oscillation with a low amplitude exists down to very low speeds. The amplitude of the oscillation increases fast with the speed near and on the other side of the bifurcation point. Such a behavior is known from stochastic dynamical systems, and probably reflects the erratic nature of the stick/slip mechanism in the dry friction damper. We also found that the amplitude of the oscillation at speeds below the bifurcation point depends on the initial condition of the dynamical problem. At speeds below the bifurcation point there exists an entire set of equilibrium solutions to the dynamical problem but in Fig. 4 only one amplitude of one representative periodic motion out of the entire set is shown.

The dynamical system was solved numerically at discrete values of a growing speed with appropriate initial conditions. An explicit Runge–Kutta 5/6th order solver with variable step length and error control was used for the integrations of the system.

The assumption of no wheel flanges or other motion limiters is of course unrealistic. True and Trepacz [36] therefore introduced a realistic wheel/rail kinematic relation in the model and repeated the investigations. The kinematic contact

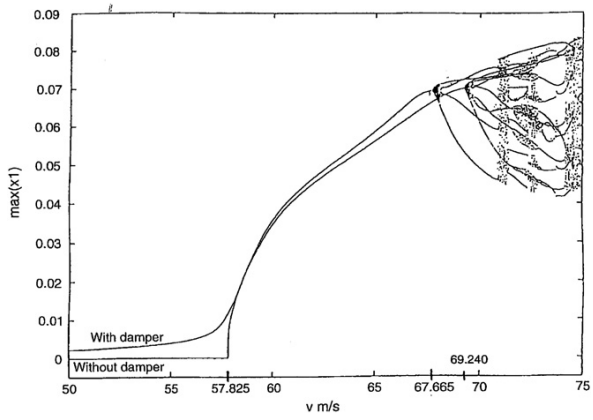


Fig. 4. Bifurcation diagrams for the single-axle bogie with and without lateral dry friction damper.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

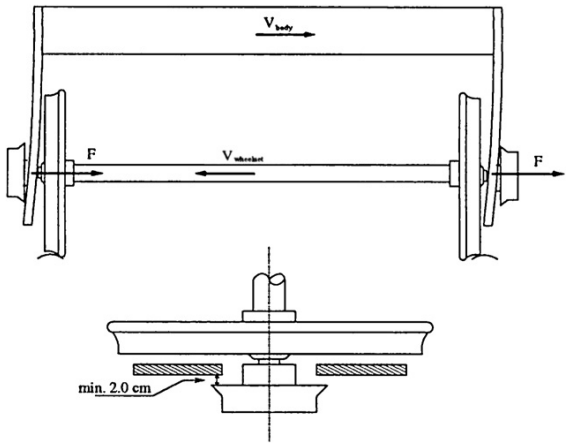


Fig. 5. The axle-guidance.

problem was solved by use of ARGE CARE's RSGEO routine [20], but again only the resultant tangent forces were taken into account in the model. In order to simplify the dynamics the horizontal component of the normal forces in the contact surface was kept constant, which of course is an unrealistic assumption.

In a real 2-axle freight wagon the motion of the axle box relative to the car body will be limited by a plate (see Fig. 5). In the lateral direction the plate acts as a linear spring with a spring constant of 1500 kN/m and a dead band of 20 mm. In the longitudinal direction the plate acts as an elastic impact with $E = 2.1 \times 10^{11}$ N/m and a dead band of 22.5 mm. E is Young's modulus for steel. This very stiff restoring force makes the dynamical system so stiff that the computation time becomes unacceptably high. We therefore approximated the impact by an ideally elastic one, where the yaw speed of the wheel set is the same before and after the impact, but its direction is reversed. We have compared some computations with either assumption and found that the dynamics remain the same, but the computation time of course increases strongly, when the impact is computed with E. If we were interested in finding the impact forces, then it would have been necessary to use the detailed model of the impact.

The limiting plate has almost no influence on the lateral dynamics, but it keeps the wheel set from derailment by limiting the maximum yaw motion. The motion is chaotic, see Fig. 6, where the maximum amplitudes of the lateral oscillations of the wheel set are plotted versus the speed of the vehicle.

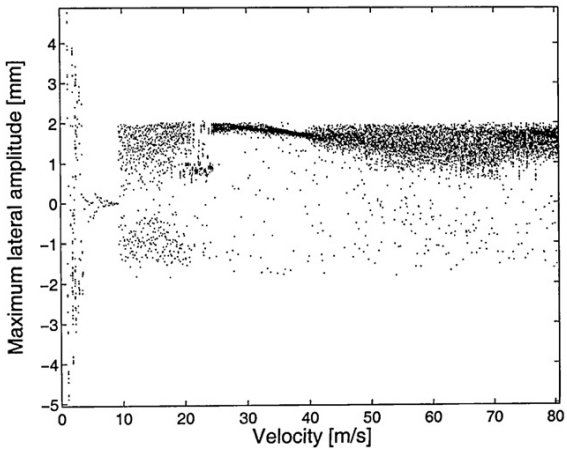


Fig. 6. Illustration of the chaotic motion of the attractor.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

Fig. 7. A 4-axle Chinese hopper wagon.

The dynamical system was solved numerically, initially with MATLAB's routine `ode45`, but then using an explicit Runge–Kutta/Cash/Karp 5/6th order solver with adaptive step size and error control. The speed of the computations with the Runge–Kutta method was around 1000 times faster than when MATLAB was used. MATLAB was, however, used for the post-processing. The time of the impact, when the yaw speed changes direction, was approximated by the time in the time stepping sequence when the axle box had penetrated the guiding plate. In the case of linear elastic impact the instants, when the axle box hit the plate and when it left the plate again, were calculated more accurately by a Newton iteration. In the time interval of the impact the forces on the axle box were supplemented by the elastic reaction forces of the plate.

5. Realistic railway vehicle models

5.1. The 4-axle hopper wagon on three-piece freight trucks

Xia and True [38,39] investigated the dynamics of a 4-axle empty Chinese hopper wagon on a straight track. The wagon (see Fig. 7) runs on two 'three-piece freight trucks' (bogies) (see Fig. 8) that are the most used bogies worldwide

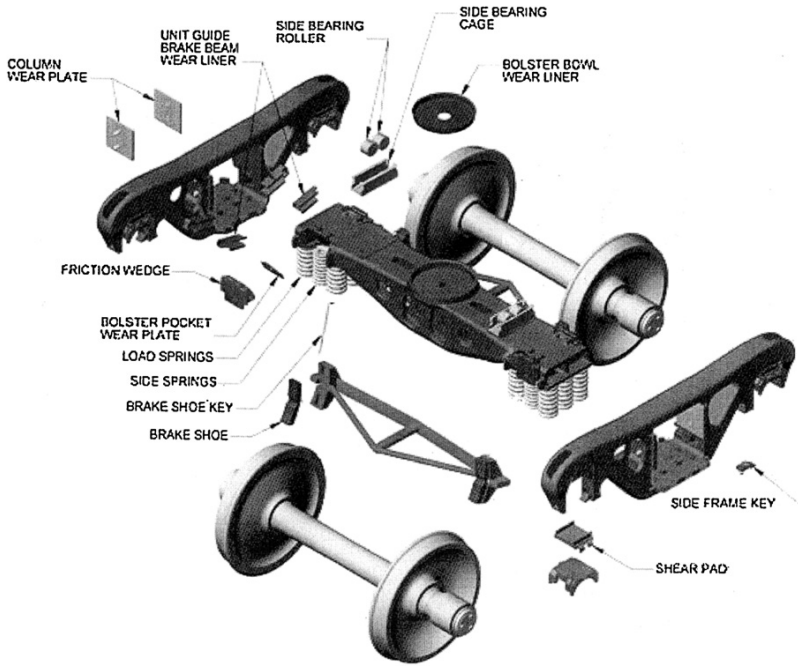


Fig. 8. Three-piece freight truck (bogie).

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

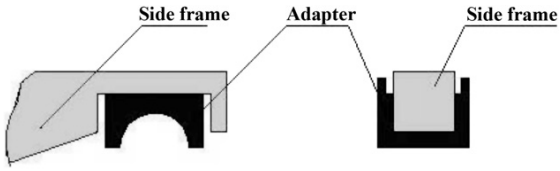


Fig. 9. The contact between the end of a frame and an adapter.

due to their simplicity, robustness and low price. The dynamics, however, leaves something to be desired. The dynamical model has 81 degrees of freedom (DOF) and is loaded with 'non-smoothnesses'. First there are the non-smoothnesses in the wheel/rail kinematic relations that we have seen earlier in this work. In addition – and that is unique for this design – all the damping is performed by dry friction with stick/slip between plane surfaces under a dynamically varying normal load. The axle boxes are fit with adapters that carry the bogie frames. The adapters can slide longitudinally under the bogie frames with dry friction contact between stops that limit their relative horizontal motion (see Fig. 9). In the only (the secondary) suspension system between the bolster and the car body (see Fig. 10) the vertical as well as the lateral damping of the relative motion are performed by dry friction with stick/slip between spring loaded wedge shaped blocks that are called 'snubbers'. Since the occurrence of stick or slip between the snubbers depends both on the normal pressure and the resulting shear force between the contacting surfaces the contact forces establish a non-smooth coupling between the horizontal and vertical components of the forces and thereby also between the horizontal and vertical dynamics. Under the influence of the dynamic forces the blocks may separate from the bolster or from the side frame, which is the source of another non-smoothness in the dynamical system. The rolling between the car body and the bogie frames is limited by bumper stops that are modelled as very stiff vertical springs with a dead band. The friction forces on the surfaces of the bumper stops are integrated into the non-smooth yaw friction torque on the car body and bolsters. Xia used the smoothened heuristic dry friction model that was used in the works in [35, Section 4]. He extended the application to two-dimensional dry friction forces on a plane. Xia introduced a friction direction angle, which replaces the sign function used in the one-dimensional dry friction analysis. The wheel/rail kinematics was calculated by his own routine WRKIN. For a description of the total model and the detailed formulation of the dynamical system the interested reader is referred to Xia's thesis [37].

Xia's main results were described in the two bifurcation diagrams in Fig. 11. The left diagram was made for growing speed and the right one for decreasing speed. The hysteresis is clearly visible. Below $V = 16$ m/s the equilibrium solutions found may be a set valued stationary motion or a combination of set valued stationary and periodic motions. A typical result for such a motion is shown in Fig. 12. At the supercritical bifurcation from the 'zero' solution on the left diagram a stable periodic solution develops. It only exists in a short speed interval after which it changes into a chaotic motion. For decreasing speed the chaotic attractor is found all the way down to 21 m/s, where it disappears – probably in a crisis. The maximum speed of the car in normal use is below 30 m/s–108 km/h. In Fig. 13 we show the chaotic lateral displacements of the four wheel sets at $V = 29$ m/s. As far as it is possible the results have been compared with tests of the dynamics of a real hopper car on a railway line, and the test results agree well with the theoretical values.

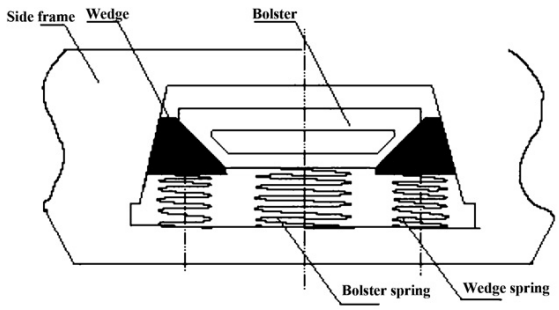


Fig. 10. A cross-section of the wedge dampers in the three-piece freight truck.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

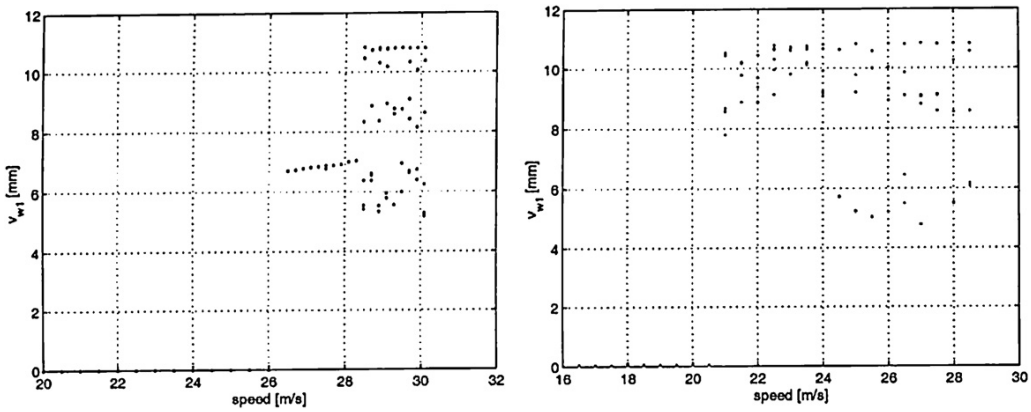


Fig. 11. Bifurcation diagrams for the Chinese hopper wagon. Left for increasing speed, right for decreasing speed.

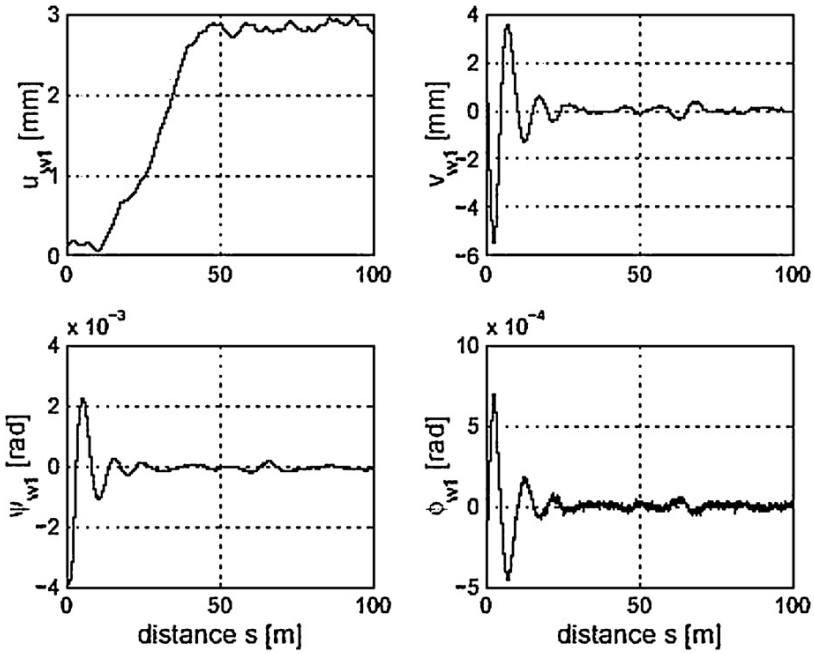


Fig. 12. The motions of the leading wheel set of the leading bogie at speed $V = 20$ m/s. Top left: the longitudinal displacement, top right: the lateral displacement, bottom left: the yaw angle and bottom right: the roll angle.

Xia used MATLAB for his calculations. The calculations were therefore very time consuming. The bifurcation diagram in Fig. 11 needed one week of shared computer time on the cluster of the DTU Informatics department(!)

The entire dynamical system with its constraint equations is a differential-algebraic system with index-3. The system was, however, transformed into an index-1 system by a differentiation with respect to time of the algebraic stick-constraint equations in the system. The index-1 system was then integrated in the domains where the state variables changed continuously by the Runge–Kutta solver `ode45` from MATLAB because it is effective. The system is stiff, and first the `ode45` solver was used, and if it failed then an implicit method was used. Due to the discontinuities each step of the integration of the system proceeded in eight steps with a loop. The details can be seen in Xia [37], where also the detailed derivation of the switch conditions are found.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

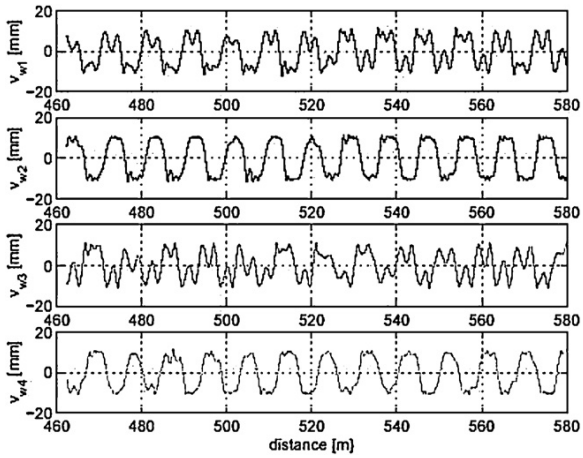


Fig. 13. The lateral displacements of the wheel sets at the speed $V = 29$ m/s as a function of the distance after the transients are negligible. From top to bottom: The leading wheel set in the leading bogie, the trailing wheel set in the leading bogie, the leading wheel set in the trailing bogie and the trailing wheel set in the trailing bogie.

Fig. 14. The Hbbills 311 wagon.

5.2. *The 2-axle freight wagon with a standard UIC-suspension*

Mark Hoffmann investigated the dynamics of two-axle European freight wagons with the UIC standard suspension [11–13]. One wagon is shown in Fig. 14, and its long wheelbase of 10 m distinguishes the wagon from the majority of two-axle wagons. The construction data were given to us from The German Railways, DB AG, in Minden. The UIC suspension (see Fig. 15) consists of two double links that connect the car body with a leaf spring that rests on an axle box. The links act as a pendulum suspension in both the lateral and longitudinal direction with combined rolling and sliding friction with stick/slip in the bearings. When the lateral displacement of a link becomes large, then the lower link will hit the bracket and the pendulum length will be halved for the further motion. The leaf spring damps the vertical motions through dry friction sliding with stick/slip between the steel leafs of the spring, and it also acts with a restoring force on the vertical motion through bending of the leafs. The mathematical model of the leaf springs that are used on the wagons was formulated by Fancher et al. [6]. The dissipated work is measured by the areas of the hysteresis loops created in the dry friction surfaces by the dynamics. Piotrowski [27] formulated the mechanical and mathematical models for the action of the links (see Fig. 16) on the basis of measurements of the behavior of a real suspension in his laboratory. They are shown in Fig. 17a and b. Piotrowski also gave values for the parameters in his models. Hoffmann has demonstrated how accurately the measured hysteresis loop in the laboratory can be approximated by Piotrowski's model when the model parameters are chosen appropriately (see Fig. 18). The wheel sets are restrained by a guidance plate with a dead band of 22.5 mm in the longitudinal and 20 mm in the lateral direction. The action of the guidances is explained in Section 4 by True and Trzepacz. Hoffmann handles the non-smoothnesses in the dynamic problem

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

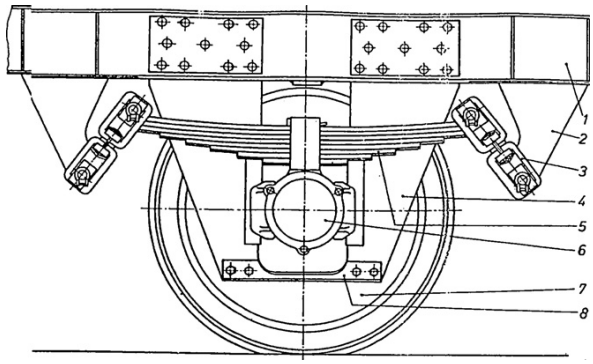


Fig. 15. The UIC standard suspension.
Reproduced from the book 'Laufwerke', Transpress, 1986.

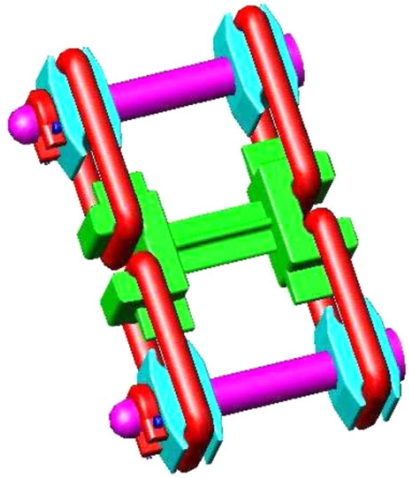


Fig. 16. The links of the UIC standard suspension.

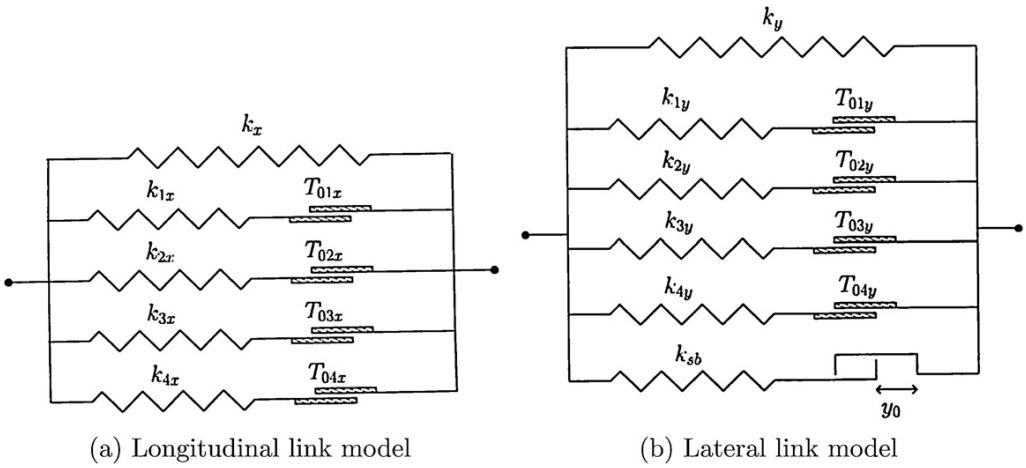


Fig. 17. The link models.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

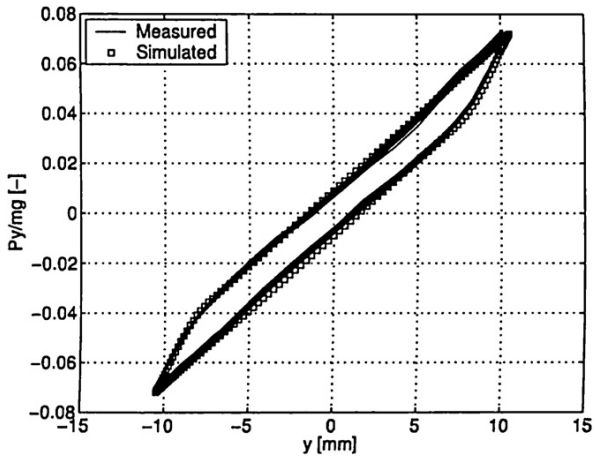


Fig. 18. A comparison between the mathematical model and the measured hysteresis loop.

through a definition of the switching boundaries and event detection. In Hoffmann's model the car body and the axles all have their own degrees of freedom, and the calculation of the instances of events when a trajectory hits a switching boundary therefore becomes much more elaborate than was the case in our earlier examples.

The non-smoothness is due to the nature of the interacting forces, i.e., stick–slip transitions in the suspension model, impacts between the axle box and axle guidance and discontinuities in the contact parameters for the wheel–rail contact. Classical solvers are all based on the existence of the derivatives of the function **F** (see Section 2). The non-smoothnesses tend to have the following effect on the numerical method: (1) the numerical solution is simply inaccurate because the progress of the solution is based on non-existing derivatives of **F**. This is a common situation for constant step size integration schemes. (2) The simulation time is unacceptably high because the step size is forced down near the non-smooth points in order to satisfy the specified error tolerance. This happens when integration schemes with variable step size and error control are applied, but it is due to the lack of smoothness of the local error. The interested reader is referred to Hoffmann's thesis [11] for a deeper discussion of the solution of this problem.

Hoffmann illustrated the importance of the location of the events. He investigated a model hysteresis loop and plotted the discrete solution points that were calculated by the ESDIRK34 NT1 solver with step and error control and event location and compared the result with the discrete solution points that were calculated with the same solver but without event location (see Fig. 19). The comparison between the figures clearly demonstrates the increase in the

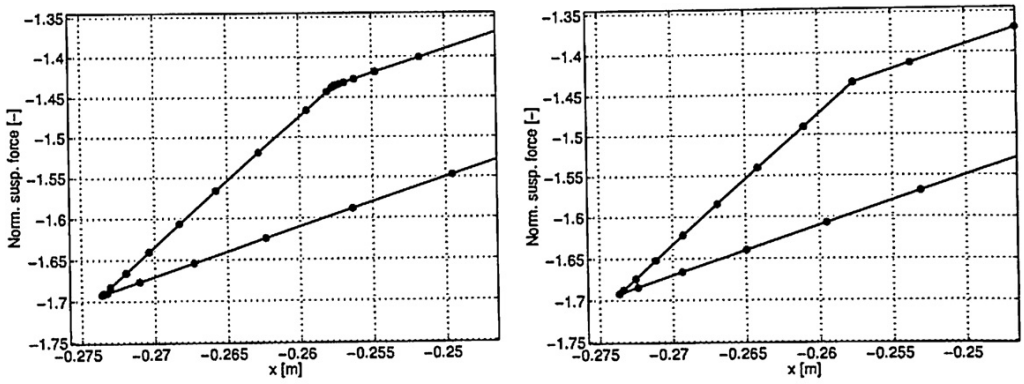


Fig. 19. The time steps on the hysteresis loop.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

number of steps without the event location, which results in a larger computational effort. It should be noted that the number of distinguishable points in the left hand corner in Fig. 19a, may be misleading, because several points may be lying so densely that the eye cannot separate them.

It is also evident from Fig. 19a that the computation time would increase enormously if a solver with constant step size had been applied. Such a solver will namely need a step size that is determined by the density of the points in the corners in order to satisfy the given error tolerance. Since the step size is constant the solver must use the same step size also in the integration along the linear sections.

Hoffmann compared the dynamics of the different types of freight wagons with UIC standard suspension. His results were presented on time series plots and bifurcation diagrams. The dynamics is very complicated with set valued stationary as well as periodic, multi-periodic and chaotic motions. He found subcritical and supercritical bifurcations into the various kinds of behavior caused by shear force instabilities and nonlinear resonances as well as symmetry breaking bifurcations. The interested reader is referred to Refs. [11–13].

The dynamical system is integrated with ESDIRK34 NT1 already mentioned in –Section 3.

The solution to the initial value problem is found by a piecewise integration strategy where each smooth section is integrated separately. The isolated events are located during the integration and treated independently. It is crucial to locate the non-smooth events during the integration. The events are determined by root finding of the event functions that define the switching boundaries between the different states of the model. For the details of the procedure the interested reader is referred to Hoffmann [11, Section 3.2].

Newton–Raphson's method needs the Jacobi matrix of the dynamical system. In our case it is a sparse matrix with $68 \times 68 = 4624$ elements of which very many are zero. Therefore the dependencies of the function $\mathbf{F}$ are identified before the integration starts, and only the non-zero elements are computed. The entries in the Jacobi matrix are computed in a column-wise fashion because the relative kinematics and interacting forces that are computed for the relative perturbations related to x_j can be reused for all non-zero elements in the jth column.

6. Discussion of numerical methods and challenges

The formulation of railway vehicle dynamical systems based on the physical principles (3) and (4) can be expressed in the form of a general initial value problem (1). In general there will be no closed form solution except the trivial state solution obtained at low speed and the models are typically both nonlinear and are subject to non-smoothness (for example in wheel–rail contact and suspension forcing). From a practitioners viewpoint, to solve such systems numerically demands the use of suitable numerical methods for the control of robustness, accuracy and efficiency. These properties are essential and without them it can be difficult to establish improved insight into the critical model behavior. A general class of numerical methods for solving (1) that have good support for local error estimation (for use with step size controllers) and event detection for non-smooth problems is the one-step/multi-stage Runge–Kutta methods.

The general class of m-stage Runge–Kutta (RK) methods for advancing (1) a single time step $\Delta t_n = t_{n+1} - t_n$ is given as

$$\begin{aligned} \mathbf{g}_i &= \mathbf{x}^n + \Delta t_n \sum_{j=1}^{m} a_{ij} \mathbf{F}(\mathrm{g_j}, t_n + c_j \Delta t_n; \mathbf{P}) \\ \mathbf{x}^{n+1} &= \mathbf{x}^n + \Delta t_n \sum_{j=1}^{m} b_j \mathbf{F}(\mathrm{g_j}, t_n + c_j \Delta t_n; \mathbf{P}) \end{aligned} \tag{16}$$

The coefficients of a convergent numerical scheme are typically given in terms of a Butcher Tableau [3] defined in terms of $\mathbf{A} \in \mathbb{R}^{m \times m}$, $\mathbf{b} \in \mathbb{R}^m$ and $\mathbf{c} \in \mathbb{R}^m$. A Runge–Kutta method is said to be order p if the local truncation error behaves asymptotically as $\mathcal{O}(\Delta t^p)$ for fixed step sizes [23]. For computations one should only use methods which have order $p > 1$ in practice due to accuracy concerns. This rules out Euler's explicit method. Local errors committed during one time step using (16) will be $\mathcal{O}(\Delta t^{p+1})$. Such errors can be estimated by comparing the computed approximate

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

solution $\mathbf{x}^{n+1}$ to one computed using an embedded Runge–Kutta method. This local error estimate can for efficiency reasons be based on the same intermediate Runge–Kutta stage values using the following formula

$$E^{(m)} = \Delta t_n \sum_{j=1}^{m} d_j \mathbf{F}(\mathrm{g_j}, t_n + c_j \Delta t_n; \mathbf{P}) \tag{17}$$

where $\mathbf{d} \in \mathbb{R}^m$. If the local error estimate is used for variable step size control, it is often possible to significantly improve efficiency over fixed step size time integration by using a variable step size controller that tries to maintain a constant accuracy level. The use of step size control has the added advantage that at the same time robustness is improved because thereby exponential growth of errors that may develop due to choices of step size will not be permitted.

The local error should be compared to user-defined acceptable error tolerances, respectively, absolute ϵ_a and relative ϵ_r levels of accuracy. In case of non-smoothness such local error estimates may become unreliable because local smoothness and asymptotic behavior of the solution is assumed. For this reason, several time steps may be rejected before the time step sizes have been reduced sufficiently for the error to be acceptable, in which case effort is wasted but the accuracy is maintained.

Dynamical systems for railway vehicles can exhibit significant stiffness due to the presence of widely different dynamical time scales in the models. For stiff systems, stability and not accuracy imposes a constraint on valid choices of the step sizes and may require significant reductions in the step sizes for securing stability. Explicit numerical schemes have bounded stability regions and therefore they may incur a performance penalty in such cases – in particular when the step size is governed by stability needs rather than accuracy. For this reason, it is customary to choose implicit solvers, which formally have large absolute (linear) stability regions. In practice, implicit Runge–Kutta methods require for each time step finding a sufficiently accurate root of the nonlinear system $\mathbf{G}(\mathbf{z}) = \mathbf{0}$ for the vector of unknown $\mathbf{z} = \left(\mathbf{g}_1^{n+1}, \ldots, \mathbf{g}_m^{n+1}\right) \in \mathbb{R}^{2Nm}$. For stiff problems this is typically done using Newton–Raphson's iterative method, which can be expressed as a two-recurrence in the compact form

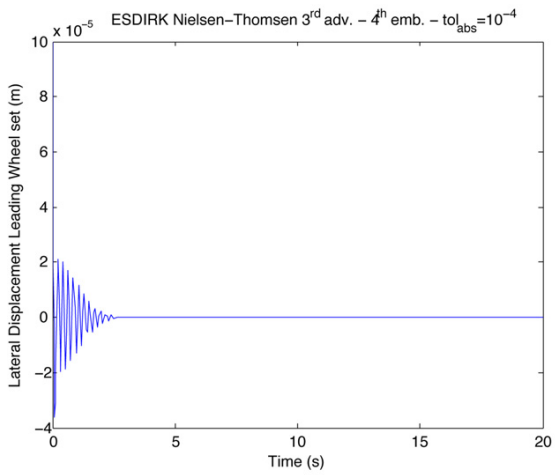
$$\mathbf{z}^{k+1} = \mathbf{z}^k + \boldsymbol{\delta}^k, \quad \boldsymbol{\delta}^k = -\mathbf{J}^{-1}\mathbf{r}^k, \quad k = 0, 1, \ldots. \tag{18}$$

where $\mathbf{J} = \partial \mathbf{G}(\mathbf{z})/\partial \mathbf{z}|_{\mathbf{z}=\mathbf{z}^k}$ is a Jacobian matrix of the system and $\mathbf{r}^k = \mathbf{G}(\mathbf{z}^k)$ is the residual of the nonlinear system in the kth iteration. For non-smooth problems, the Jacobian matrix can be singular or ill-conditioned in a point and this can be the cause of numerical problems if event detection is not used [11]. Reduction in solution effort per time step of the Runge–Kutta method is typically achieved by exploiting properties in the coefficients of $\mathbf{A}$ and/or using an inexact constant approximation to $\mathbf{J}$ in the inner solve step for determining $\boldsymbol{\delta}^k \approx \mathbf{z}^k - \mathbf{z}$. However, this may be at the expense of slowed down convergence rates and a resulting decrease in algorithmic efficiency which needs to be balanced by improved numerical efficiency. A class of Runge–Kutta schemes that is subject to the idea of minimizing the work effort per step and also have good stability properties are the ESDIRK methods. A suitable stopping criterion for the Newton–Raphson method is based on making sure that a measure of the estimated errors $\boldsymbol{\delta}^k$ in the inner loop of each Runge–Kutta stage is sufficiently small for the errors committed in one complete Runge–Kutta step to be dominant.

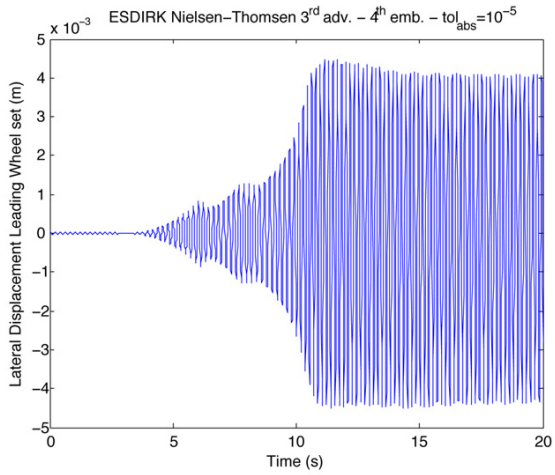
A number of pre-packaged scientific solvers for the solution of systems of ordinary differential equations exist (e.g., see http://www.netlib.org) but details will not be given. However, they will be referenced in the following where appropriate.

Performance is another key concern for practical use of solvers. It can be useful to evaluate the performance of a numerical scheme in terms of algorithmic and numerical efficiencies. The algorithmic efficiency is measured in terms of iteration counts (successful/failed steps and function evaluations), and the numerical efficiency is a direct measure of wall clock time. To compare alternative methods the step size history needs to be taken into account and a fair comparison can be done by using the same step size control for each method together with a specification of the same acceptable tolerance level. As an example, a recent investigation of the dynamics of the Cooperrider's bogie model shown in Fig. 2 has been performed on a straight track at $V = 40$ m/s (not hunting) and $V = 120$ m/s (hunting) using two different RK methods with same step size control and different tolerance levels. In Fig. 20 we present computed results obtained with the package SDIRK [25] which includes a PI step size control strategy (e.g., see [8]). The basic version of the package contains the ESDIRK34 NT1 method by Nielsen–Thomsen [24] and the code has been extended to include an Explicit Runge–Kutta–Fehlberg method ERKF34 [9], for use in combination with the existing PI controller to make comparisons fair. A detailed breakdown of important performance characteristics is given in Table 1. It is

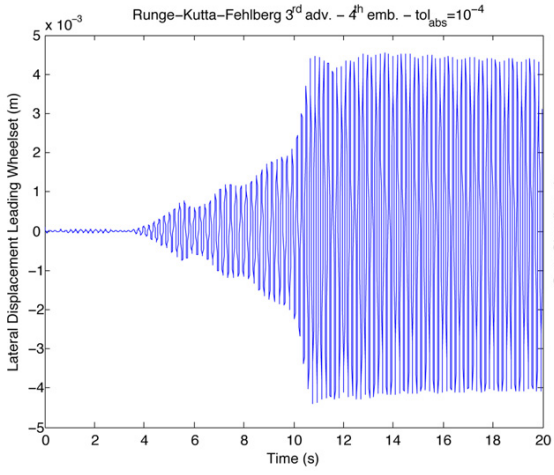
Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016



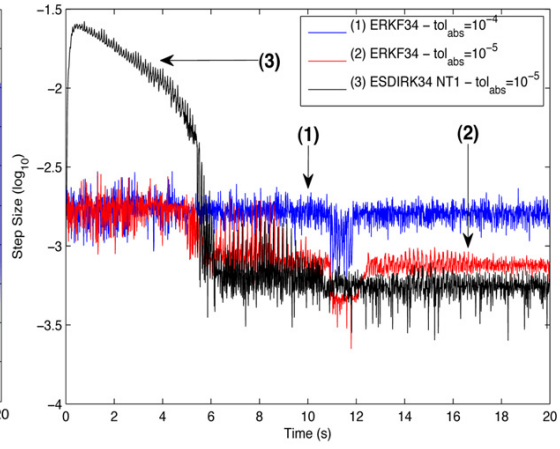
(a) ESDIRK34 NT1 ($\epsilon_a = 10^{-4}$)



(b) ESDIRK34 NT1 ($\epsilon_a = 10^{-5}$)



(c) ERKF34 ($\epsilon_a = 10^{-4}$)



(d) Step size histories

Fig. 20. Computed results for lateral displacement of leading wheel set of Cooperrider's bogie model for different user-defined absolute tolerance levels. Using ESDIRK34 NT1 it is found that (a) the transient behavior is fully damped for $\epsilon_a = 10^{-4}$ and (b) periodic oscillations (hunting) are captured for $\epsilon_a = 10^{-5}$. With ERKF34 it is found that (c) periodic oscillations (hunting) are captured already at $\epsilon_a = 10^{-4}$.

Table 1
Performances of the RKF34 and ESDIRK34 NT1 for solving a transient analysis of 20 s of a Cooperrider model hunting. The table shows the absolute tolerance used, the method's names, the wall clock time, the number of function evaluations, the number of Jacobian evaluations, the number of accepted steps and the number of rejected steps. ESDIRK34 NT1 with tolerance 10^{-4} fails in detecting the hunting phenomenon.

ε_a	Solver	CPU time	# Fun. ev.	# Jac. ev.	# Acc.	# Rej.
10^{-4}	ERKF34	15.34 s	74,505		12,617	6009
	ESDIRK34 NT1	1.47 s	1181	112	90	20
10^{-5}	ERKF34	21.25 s	130,389		22,989	9613
	ESDIRK34 NT1	467.12 s	428,957	34,911	25,525	9385

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

noticeable that the hunting phenomenon can be captured using the explicit ERKF34 but not the implicit ESDIRK34 at a tolerance level $\epsilon_a = 10^{-4}$. The reason is that the implicit method exhibits strong numerical damping of these high frequency modes at this tolerance level. With a reduced tolerance level $\epsilon_a = 10^{-5}$ the implicit ESDIRK34 has reduced numerical damping of the hunting modes and captures the phenomenon. However, it has a wall clock time which is close to 22 times larger as a result of more work per step compared to the explicit solver for this tolerance level. This result challenges the wide use of implicit methods instead of explicit methods. It also highlights the importance of tuning the (usually user-defined) tolerance level to be able to resolve a physical phenomenon of interest. It demonstrates that explicit solvers from a performance viewpoint can be more attractive for both efficient and accurate analysis than an alternative implicit method of similar formal accuracy.

7. Lessons learned

It is highly recommended to employ numerical schemes for dynamic railway vehicle simulations which employ variable step size control for control of local errors (targets efficiency, robustness and accuracy), introduce the relevant switching boundaries in the model formulations (targets accuracy and efficiency) and make use of event location for the numerical solution of non-smooth dynamical problems (targets accuracy and efficiency). Final results should be subject to convergence tests to rule out the possibility of errors, which may arise from the choice of too relaxed tolerance levels. For numerical investigation of chaotic dynamics we have experienced that explicit solvers may have an advantage over implicit solvers, because for accurate results the step size is bound by accuracy rather than stability requirements and the explicit methods require less work per step for same formal order of accuracy.

The time spent with the formulation of the root finding method for determination of the events and of the laws that apply in the events is a cheap investment in a numerical routine that then will operate much faster and yield reliable results. If however the switching boundaries lie very close together in the state space other strategies may apply, see e.g., Studer and Glocker [31], where a modified scheme is applied.

References

[1] M. Arnold, B. Burgermeister, C. Führer, G. Hippmann, G. Rill, Numerical methods in vehicle system dynamics: state of the art and current developments, Vehicle System Dynamics 49 (2011) 1159–1207.
[2] D. Bigoni, Curving Dynamics in High Speed Trains, Master's thesis, IMM, The Technical University of Denmark, Kongens Lyngby, Denmark, 2011.
[3] J. Butcher, Numerical Methods for Ordinary Differential Equations, J. Wiley, Chichester, West Sussex, England, Hoboken, NJ, 2003.
[4] N. Cooperrider, The hunting behavior of conventional railway trucks, ASME Journal of Engineering and Industry 94 (1972) 752–762.
[5] W.H. Enright, K.R. Jackson, S.P. Nørsett, P.G. Thomsen, Interpolants for Runge–Kutta formulas, ACM Transactions on Mathematical Software 12 (1986) 193–218.
[6] P.S. Fancher, R.D. Ervin, C.C. MacAdam, C.B. Winkler, Measurement and representation of the mechanical properties of truck leaf springs, SAE Transactions (1980).
[7] V.K. Garg, R.V. Dukkipati, Dynamics of Railway Vehicle Systems, Academic Press, Toronto, Orlando, San Diego, New York, London, Montreal, Sydney, Tokyo, 1984.
[8] K. Gustafsson, Control of Error and Convergence in ODE Solvers, Ph.D. thesis, Department of Automatic Control, Lund Institute of Technology, 1992.
[9] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations I: Nonstiff problems, Solving Ordinary Differential Equations I: Nonstiff Problems, Springer Series in Computational Mathematics, second revision ed., Springer-Verlag, Berlin, Wien, New York, 1991.
[10] H. Hertz, Über die Berührung fester elastischer Körper, Journal für die Reine und Angewandte Mathematik 92 (1881) 156–171.
[11] M. Hoffmann, Dynamics of European Two-axle Freight Wagons, Ph.D. thesis, IMM, The Technical University of Denmark, http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=4853, 2006.
[12] M. Hoffmann, H. True, On the dynamics of a railway freight wagon with UIC standard suspension, in: I. Zobory (Ed.), Proc. 9th Miniconf. on Vehicle System Dynamics, Identification and Anomalies, Budapest, November 8–10, 2004, Budapest University of Technology and Economics, Budapest, Hungary, 2006, pp. 91–98.
[13] M. Hoffmann, H. True, The dynamics of european two-axle railway freight wagons with UIC standard suspension, in: J.K. Hedrick (Ed.), Proc. 20th IAVSD Symposium of The International Association for Vehicle System Dynamics, Taylor & Francis, 2008, pp. 225–236.
[14] P. Isaksen, H. True, On the ultimate transition to chaos in the dynamics of cooperrider's bogie, Chaos, Solitons and Fractals 8 (1997) 559–581.
[15] S. Iwnicki (Ed.), The Manchester Benchmarks for Rail Vehicle Simulation, Vehicle System Dynamics, vol. 31, Supplement, Swets & Zeitlinger, Lisse, 1999.
[16] C.N. Jensen, H. True, On a new route to chaos in railway dynamics, Nonlinear Dynamics 13 (1997) 117–129.
[17] C. Kaas-Petersen, Chaos in a railway bogie, Acta Mechanica 61 (1986) 89–107.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

[18] C. Kaas-Petersen, PATH – User's Guide, Technical Report, Department of Applied Mathematical Studies and Centre for Nonlinear Studies, University of Leeds, 1989.
[19] J. Kalker, wheel–rail rolling contact theory, Wear 144 (1991) 243–261.
[20] W. Kik, D. Moelle, ACRadSchiene – To create or Approximate Wheel/Rail profiles, Technical Report, 2010.
[21] K. Knothe, F. Böhm, History of stability of railway and road vehicles, Vehicle System Dynamics 31 (1999) 283–323.
[22] C. Knudsen, R. Feldberg, H. True, Bifurcations and chaos in a model of a rolling wheelset, Philosophical Transactions of the Royal Society A 338 (1992) 455–469.
[23] R. Leveque, Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems (Classics in Applied Mathematics), SIAM, Society for Industrial and Applied Mathematics, Philadelphia, USA, 2007.
[24] H.B. Nielsen, P.G. Thomsen, Hæfte 66 – Numeriske Metoder for Sædvanlige differentialligninger, Numerisk Institut, DTH, 1993.
[25] E. Østergaard, Documentation for the SDIRK C++Solver, Technical Report 2, IMM, Technical University of Denmark, 1998.
[26] L. Petzold, Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations, SIAM Journal on Scientific and Statistical Computing 4 (1983) 136–148.
[27] J. Piotrowski, Model of the UIC link suspension for freight wagons, Archive of Applied Mechanics 73 (2003) 517–532.
[28] G. Sauvage, J.P. Pascal, Solution of the multiple wheel and rail contact dynamic problem, Vehicle System Dynamics 19 (1990) 257–272.
[29] Z.Y. Shen, J.K. Hedrick, J.A. Elkins, A comparison of alternative creep-force models for rail vehicle dynamic analysis, in: J.K. Hedrick (Ed.), The Dynamics of Vehicles, Proc. 8th IAVSD Symp., Cambridge, MA, Swets and Zeitlinger, Lisse, 1984, pp. 591–605.
[30] E. Slivsgaard, H. True, Chaos in Railway-vehicle Dynamics, Nonlinearity and Chaos in Engineering Dynamics, John Wiley & Sons Ltd, Chichester, 1994, pp. 183–192.
[31] C. Studer, C. Glocker, Simulation of non-smooth mechanical systems with many unilateral constraints, EUROMECH Newsletter 29 (May 2006) 15–33.
[32] P.G. Thomsen, H. True (Eds.), Non-smooth Problems in Vehicle Systems Dynamics, Proc. Euromech 500 Colloquium, Springer, Berlin, Wien, New York, 2010.
[33] H. True, On a new phenomenon in bifurcations of periodic orbits, in: Dynamics, Bifurcation and Symmetry, New Trends and New Tools, September 3–9, 1993, Kluwer Academic Publishers, P.O. Box 322, NL-3300 AH Dordrecht, The Netherlands, 1994, pp. 327–331.
[34] H. True, Dynamics of railway vehicles and rail/wheel contact, Dynamics of Railway Vehicles and Rail/Wheel Contact, CISM Courses and Lectures – No. 497, Springer, Wien, New York, 2007, pp. 75–128.
[35] H. True, R. Asmund, The dynamics of a railway freight wagon wheelset with dry friction damping, Vehicle System Dynamics 38 (2002) 149–163.
[36] H. True, L. Trzepacz, The dynamics of a railway freight wagon wheelset with dry friction damping in the suspension, in: Proc. 18th IAVSD Symposium on Vehicle System Dynamics, The Dynamics of Vehicles on Roads and Tracks, Taylor & Francis, London, UK, 2004, pp. 587–596.
[37] F. Xia, The Dynamics of The Three-Piece-Freight Truck, Ph.D. thesis, IMM, The Technical University of Denmark, http://www2.imm.dtu.dk/pubdb/public/search.php?searchstr=Fujie+Xia&n=5&searchtype=strict, 2002.
[38] F. Xia, H. True, On the dynamics of the three-piece-freight truck, in: RTD, vol. 25, IEEE/ASME Joint Rail Conference, Chicago, IL, April 22–24, 2003, American Society of Mechanical Engineers, United Engineering Center, 345 East 47th Street, New York, NY 10017, USA, 2003, pp. 149–159.
[39] F. Xia, H. True, The dynamics of the three-piece-freight truck, in: Proc. 18th IAVSD Symposium on Vehicle System Dynamics, The Dynamics of Vehicles on Roads and on Tracks, Taylor & Francis, London, UK, 2004, pp. 212–221.

Please cite this article in press as: H. True, et al. On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics, Math. Comput. Simul. (2012), http://dx.doi.org/10.1016/j.matcom.2012.09.016

Noname manuscript No.
(will be inserted by the editor)

A Stochastic Nonlinear Water Wave Model for Efficient Uncertainty Quantification

Daniele Bigoni · Allan P. Engsig-Karup · Claes Eskilsson

arXiv:1410.6338v1 [physics.comp-ph] 23 Oct 2014

Received: date / Accepted: date

Abstract A major challenge in next-generation industrial applications is to improve numerical analysis by quantifying uncertainties in predictions. In this work we present a stochastic formulation of a fully nonlinear and dispersive potential flow water wave model for the probabilistic description of the evolution waves. This model is discretized using the Stochastic Collocation Method (SCM), which provides an approximate surrogate of the model. This can be used to accurately and efficiently estimate the probability distribution of the unknown time dependent stochastic solution after the forward propagation of uncertainties. We revisit experimental benchmarks often used for validation of deterministic water wave models. We do this using a fully nonlinear and dispersive model and show how uncertainty in the model input can influence the model output. Based on numerical experiments and assumed uncertainties in boundary data, our analysis reveals that some of the known discrepancies from deterministic simulation in comparison with experimental measurements could be partially explained by the variability in the model input. This type of stochastic analysis is relevant for computationally intensive problems where traditional methods, such as Monte Carlo type methods, are intractable due to their slow convergence. The Stochastic Collocation Method exhibits faster convergence and retains the non-intrusive properties of Monte Carlo methods, allowing for the straight forward use of massively parallel computing.

Keywords Uncertainty Quantification · generalized Polynomial Chaos · heterogeneous computing · high-performance computing · free surface water waves · Laplace problem · partial differential equations.

D. Bigoni · A.P. Engsig-Karup
Department of Applied Mathematics and Computer Science, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark
E-mail: dabi@dtu.dk

C. Eskilsson
Department of Shipping and Marine Technology, Chalmers University of Technology, SE–412 96 Gothenburg, Sweden

1 Introduction

In coastal and offshore engineering it is important to design maritime structures that can withstand critical failures due to wave-induced loadings. The most extreme wave induced-loadings can be estimated from direct measurements, laboratory experiments and simulation-based tools which can account for the wave kinematics sufficiently accurately. It is still common to predict wave kinematics using numerical tools which have been validated by single or few deterministic simulations and compared to idealized physical experiments, e.g., in wave tanks. In an era with fast growing computing power, computational simulation tools are increasingly being used for engineering studies and analysis. In particular, this trend is driven by improvements in hardware which have seen a recent paradigm shift from single to many core computations. Parallel to this shift, there is an increased focus on making simulation tools more reliable by estimation and reduction of uncertainty in results delivered by such tools. This requires a shift from deterministic approaches to probabilistic approaches [35]. This is of immense importance for tools that are used for critical decision support, risk management and risk analysis.

The research field of Uncertainty Quantification deals with mathematical techniques that can improve engineering analysis in model-based simulation tools. The goal is to deliver confidence intervals and estimation of probability distributions of Quantities of Interest (QoIs) to describe the likelihood for the QoIs to take a given value. The analysis of uncertainty in dynamical systems can be split in four steps:

(a) Deterministic modeling and identification of Quantities of Interest (QoI) and sources of uncertainty
(b) Quantification of uncertainty sources by means of probability distributions
(c) Uncertainty propagation through the system
(d) Sensitivity analysis

This work will deal with the first three of these steps, where classical benchmarks, such as [1,10], will be used as deterministic models and different QoI will be investigated for the different problems (step a). In coastal and offshore engineering, the QoIs are typically local wave statistics for average or maximum heights, loads, etc. The analysis of such QoIs is useful for risk management aimed at reducing risk in design and operations. For example, in structural engineering Ultimate Limits State (ULS) design are today based on load and resistance factors determined using statistics obtained based on measured data or experiments.

Due to the lack of data, some assumptions will be made about the probability distributions of the sources of uncertainty (step b), that in hydrodynamics simulations are commonly inlet/outlet conditions (boundary conditions), bathymetry data and structural positions (geometry). All these uncertainties can be classified as *epistemic* [19], because they can in principle be reduced either by better measurements and/or, in case of experimental tests, by more accurate settings. The step (c) will be the main focus of this work, where the

propagation of the probability distributions through the dynamical system will be investigated. Traditional sampling techniques, such as Monte Carlo methods, will be compared to modern techniques based on generalized Polynomial Chaos [36]. Non-intrusive approaches such as Stochastic Collocation and Sparse Grids will be preferred to intrusive approaches, due to the ability of the former of re-using existing code, avoiding the need for re-engineering existing software. The step (d) concerns with the identification of the sources of uncertainty that give the biggest contribution to the uncertainty of the QoI. This topic will not be covered here, but its application is based on all the techniques used in (c).

Uncertainty quantification in coastal and offshore engineering is challenged by the requirements of computational resources for single deterministic simulations. Recent works [27] have explored the usage of Monte Carlo type methods for the estimation of extreme responses. However, these methods show a very slow convergence rate and even with the disruptive introduction of many-core hardware and parallel simulation tools [13,12,17], they become quickly intractable, because few simulations are affordable in general. Thus, techniques with fast convergence, such as the Stochastic Collocation Method become very important in this setting as well as in many other engineering areas dominated by heavy computational requirements.

1.1 Paper contributions

We propose a stochastic formulation of a fully nonlinear and dispersive potential flow model for efficient uncertainty quantification. We revisit classical benchmarks and propose to use the stochastic collocation method for ensuring that the ensemble of solutions can be generated independently using standard deterministic solvers as black-box methods with tunable parameters. The outcome is a set of stochastic benchmarks. The analysis reveals opportunities and challenges in practical uncertainty quantification that needs to be addressed for computationally intensive computer simulation and engineering analysis.

1.2 Paper organization

The paper will be organized as follows. In Section 2 we introduce the governing equation for the deterministic description of nonlinear water waves based on potential theory. In Section 3 we describe how a stochastic model can be formulated, including a description of the Stochastic Collocation Method (SCM) approach for creating approximate generalized Polynomial Chaos (gPC) surrogate models of the solutions. In Section 4, the effect of parametric uncertainty in bathymetry and wave input are studied and numerical experiments are compared for traditional sampling and SCM approaches.

2 Mathematical formulation

We consider unsteady water waves described by a potential model for three-dimensional fully nonlinear and dispersive free surface flows under the influence of gravity. The flow is assumed inviscid and irrotational. It can, without simplifications, be used for short and long wave propagation in both shallow and deep water where viscous and rotational effects are negligible. The sea bed is assumed variable and impermeable.

We introduce a Cartesian coordinate system (x, y, z) with (x, y) the horizontal and z the vertical dimensions, where the z coordinate points upwards. The functions $h(x, y)$ and $\zeta(t, x, y)$ describe respectively the depth of the sea bed and the free surface. The still water level is given by $z = 0$.

2.1 The deterministic model

The evolution of water waves over an arbitrary sea bed are described by the kinematic and dynamic free surface boundary conditions[1]

$$\partial_t \zeta(\mathbf{x}, t) = -\boldsymbol{\nabla}\zeta \cdot \boldsymbol{\nabla}\tilde{\phi} + \tilde{w}(1 + \boldsymbol{\nabla}\zeta \cdot \boldsymbol{\nabla}\zeta), \tag{1a}$$

$$\partial_t \tilde{\phi}(\mathbf{x}, t) = -g\zeta - \frac{1}{2}\left(\boldsymbol{\nabla}\tilde{\phi} \cdot \boldsymbol{\nabla}\tilde{\phi} - \tilde{w}^2(1 + \boldsymbol{\nabla}\zeta \cdot \boldsymbol{\nabla}\zeta)\right), \tag{1b}$$

where $\boldsymbol{\nabla} = (\partial_x, \partial_y)$. We will consider waves in a spatial domain $D \in \mathbb{R}^l$ (fluid volume), $l = 2, 3$ and a time domain $t \in [0, T]$ with final time $T > 0$. For the fluid volume, a Laplace problem defines the scalar velocity potential

$$\phi = \tilde{\phi}, \quad z = \zeta(\mathbf{x}, t), \tag{2a}$$

$$\boldsymbol{\nabla}^2\phi + \partial_{zz}\phi = 0, \quad -h \le z < \zeta(\mathbf{x}, t), \tag{2b}$$

$$\partial_z\phi + \boldsymbol{\nabla}h \cdot \boldsymbol{\nabla}\phi = 0, \quad z = -h. \tag{2c}$$

Using a classical σ-transformation

$$\sigma \equiv \frac{z + h(\boldsymbol{x})}{d(\boldsymbol{x}, t)}, \quad 0 \le \sigma \le 1, \tag{3}$$

the Laplace problem can be written as

$$\Phi = \tilde{\phi}, \quad \sigma = 1, \tag{4a}$$

$$\begin{aligned}\boldsymbol{\nabla}^2\Phi + \boldsymbol{\nabla}^2\sigma(\partial_\sigma\Phi) + 2\boldsymbol{\nabla}\sigma \cdot \boldsymbol{\nabla}(\partial_\sigma\Phi) + \\ (\boldsymbol{\nabla}\sigma \cdot \boldsymbol{\nabla}\sigma + (\partial_z\sigma)^2)\partial_{\sigma\sigma}\Phi = 0, \quad 0 \le \sigma < 1,\end{aligned} \tag{4b}$$

$$\mathbf{n} \cdot (\boldsymbol{\nabla}, \partial_z\sigma\partial_\sigma)\Phi = 0, \quad (\mathbf{x}, \sigma) \in \partial\Omega, \tag{4c}$$

[1] The gravitational acceleration constant, g, is set to be 9.81 m/s^2.

where

$$\boldsymbol{\nabla}\sigma = \tfrac{1-\sigma}{d}\boldsymbol{\nabla}h - \tfrac{\sigma}{d}\boldsymbol{\nabla}\zeta, \tag{5a}$$

$$\boldsymbol{\nabla}^2\sigma = \tfrac{1-\sigma}{d}\left(\boldsymbol{\nabla}^2 h - \tfrac{\boldsymbol{\nabla}h\cdot\boldsymbol{\nabla}h}{d}\right) - \tfrac{\sigma}{d}\left(\boldsymbol{\nabla}^2\zeta - \tfrac{\boldsymbol{\nabla}\zeta\cdot\boldsymbol{\nabla}\zeta}{d}\right) - \tfrac{1-2\sigma}{d^2}\boldsymbol{\nabla}h\cdot\boldsymbol{\nabla}\zeta - \tfrac{\boldsymbol{\nabla}\sigma}{d}\cdot\left(\boldsymbol{\nabla}h + \boldsymbol{\nabla}\zeta\right), \tag{5b}$$

$$\partial_z\sigma = \tfrac{1}{d}. \tag{5c}$$

The relation between the scalar velocity potential function and velocity field is

$$(\mathbf{u}, w) = (\boldsymbol{\nabla} + \boldsymbol{\nabla}\sigma\partial_\sigma, \partial_z\sigma\partial_\sigma)\Phi. \tag{6}$$

The governing equations can be solved in the setting of a numerical wave tank and is then subject to initial and boundary conditions

$$\zeta(\mathbf{x}, t=0) = \phi(\mathbf{x}, t=0) = 0, \quad \partial_n\zeta = \partial_n\phi = 0, \quad x \in \partial D\backslash\bar{D}^{FS}, \tag{7}$$

where wave generation and absorption is done using a line relaxation method [21]. A complete derivation of the equations are given in [12]. These model equations can be solved numerically using flexible-order finite differences [11, 13] and the massively parallel implementation [17] enables fast hydrodynamics computations [12]. A fast solver is a prerequisite for enabling stochastic analysis with acceptable time frames and can be used to deliver improved engineering analysis in maritime applications.

2.2 The stochastic model

Following [36], a stochastic formulation is obtained by introducing $\omega \in \Omega$ as random input of the system defined in the probability space $(\Omega, \mathcal{F}, \mathcal{P})$, where Ω is the sample space, $\mathcal{F}$ is a σ-field and $\mathcal{P}$ is a probability measure. This makes the unknown solution a random process $\zeta(\mathbf{x}, t, \omega) : \bar{D}^{FS} \times [0, T] \times \Omega \to \mathbb{R}$ and $\phi(\mathbf{x}, t, \omega) : \bar{D} \times [0, T] \times \Omega \to \mathbb{R}$. $\bar{D}$ is the closed spatial domain volume with FS indicating the restriction to the free surface, $\bar{D} = \{\mathbf{x}|\mathbf{x} \in \xi\}$.

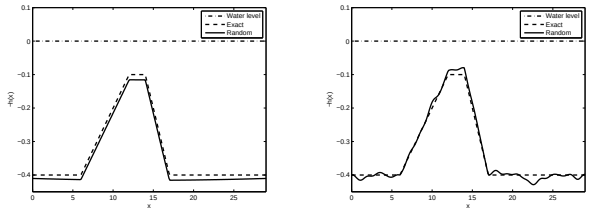
A parametrization of the stochastic model is required in order to solve it numerically. A set of random variables $\mathbf{Z} : \Omega \to \mathbb{R}^d$, is introduced to characterize random inputs, where $d \geq 1$ is the stochastic dimension.

The stochastic reformulation of the deterministic system (1) is

$$\partial_t\zeta(\mathbf{x}, t, \mathbf{Z}) = -\boldsymbol{\nabla}\zeta\cdot\boldsymbol{\nabla}\tilde{\phi} + \tilde{w}(1 + \boldsymbol{\nabla}\zeta\cdot\boldsymbol{\nabla}\zeta), \tag{8a}$$

$$\partial_t\tilde{\phi}(\mathbf{x}, t, \mathbf{Z}) = -g\zeta - \frac{1}{2}\left(\boldsymbol{\nabla}\tilde{\phi}\cdot\boldsymbol{\nabla}\tilde{\phi} - \tilde{w}^2(1 + \boldsymbol{\nabla}\zeta\cdot\boldsymbol{\nabla}\zeta)\right), \tag{8b}$$

where for any realization of an uncertain sea state, the Laplace problem (2) is fulfilled to obtain closure.



(a) Random field ($a = 1.0$) (b) Random field ($a = 0.1$)

Fig. 1: Possible topographies of the bottom floor in the submerged bar experiment. Figures 1a and 1b show two realizations of the KL-expanded random fields (9) with different correlation lengths $a = 1$ and $a = 0.1$ respectively, where the total variance represented is ≥ 0.95.

2.3 Dimensionality reduction using the Karhunen-Loève Expansion

As an example, in the wave model, the bathymetry function describing still-water depth can be uncertain and therefore be treated as a random field. The Karhunen-Loève expansion (KLE) is a useful technique for dimension reduction that can be used for the parametrization of such random fields [22, 30]. Let $h(\mathbf{x}, \omega)$ be a spatially varying random field over a spatial domain Ω with mean $\mu_h(\mathbf{x})$ and covariance function $C(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{Cov}(h(\mathbf{x}_1, \omega), h(\mathbf{x}_2, \omega))$. Then the bathymetry function $h(\mathbf{x}, \omega)$ can be parametrized as an infinite series

$$h(\mathbf{x}, \omega) = \mu_h(\mathbf{x}) + \sum_{i=1}^{\infty} \sqrt{\lambda_i} \psi_i(\mathbf{x}) Y_i(\omega), \tag{9}$$

where $\mathbf{E}[Y_i(\omega)] = 0$, $\mathbf{Cov}[Y_i, Y_j] = \delta_{ij}$ and $\{\lambda_i, \psi_i\}_{i=1}^{\infty}$ are the solutions of the generalized eigenvalue problem

$$\int_{\Omega} C(\mathbf{x}, \mathbf{s}) \psi_i(\mathbf{s}) d\mathbf{s} = \lambda_i \psi_i(\mathbf{x}). \tag{10}$$

If $h(\mathbf{x}, \omega)$ is a Gaussian random field, then $Y_i \sim \mathcal{N}(0, 1)$.

For practical computations (9) is truncated at a desired order N. It is easy to check how much of the variance of the original random field is retained by such approximation, using that

$$\begin{aligned} \mathbf{Var}\left[h_N(\mathbf{x}, \omega)\right] &= \mathbf{E}\left[h_N^2(\mathbf{x}, \omega)\right] - \mathbf{E}\left[h_N(\mathbf{x}, \omega)\right]^2 \\ &= \mathbf{E}\left[\sum_{i,j=1}^{N} \sqrt{\lambda_i \lambda_j} \psi_i(\mathbf{x}) \psi_j(\mathbf{x}) Y_i(\omega) Y_j(\omega)\right] = \sum_{i=1}^{N} \lambda_i \psi_i^2(\mathbf{x}), \end{aligned}$$

where the orthogonality of $\{Y_i\}_{i=1}^{N}$ is exploited. There are several options regarding the correlation kernel $C(\mathbf{x}_1, \mathbf{x}_2)$. All these are problem dependent and

an appropriate characterization of the random field has to be performed prior to the construction of the KL-expansion. In this work, we will use the exponential covariance kernel

$$C(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|}{a}\right), \tag{11}$$

where a is the correlation length. Figure 1 shows realizations of the KL-expansions of a 1D random field $h(x, \omega)$ for the submerged bar experiment considered in section 4.1 with exponential covariance kernel and zero mean for different correlation length a. The total variance represented by the KL-expansions $h_N(x, \omega)$ is fixed to 0.95 (the total variance of $h(x, \omega)$ with exponential covariance kernel is 1). In Figure 1a and 1b, fields with different correlation lengths are illustrated. Shorter correlation lengths determine a slower decay of the expansion coefficients in (9) and thus a longer expansion is required to express higher local variability.

3 Uncertainty Quantification

In Uncertainty Quantification we are interested in studying the propagation of uncertainties through the stochastic dynamical system (8). To reduce the notation used, let $\mathbf{u}(\mathbf{x}, t, \mathbf{Z}) = [\zeta(\mathbf{x}, t, \mathbf{Z}), \tilde{\phi}(\mathbf{x}, t, \mathbf{Z})]^T$. We are interested in describing the stochastic result in terms of its probability distribution and/or its first moments, e.g., mean and variance

$$\mathbf{E}[\mathbf{u}(\mathbf{x}, t, \mathbf{z})] = \int_{\mathbb{R}^d} \mathbf{u}(\mathbf{x}, t, \mathbf{z}) dF_{\mathbf{z}}(\mathbf{z}) = \int_{\mathbb{R}^d} \mathbf{u}(\mathbf{x}, t, \mathbf{z}) \rho_{\mathbf{z}}(\mathbf{z}) d\mathbf{z} = \mu_{\mathbf{u}}(\mathbf{x}, t), \tag{12}$$

$$\mathbf{Var}[\mathbf{u}(\mathbf{x}, t, \mathbf{z})] = \int_{\mathbb{R}^d} \left(\mathbf{u}(\mathbf{x}, t, \mathbf{z}) - \mu_{\mathbf{u}}(\mathbf{x}, t)\right)^2 \rho_{\mathbf{z}}(\mathbf{z}) d\mathbf{z}, \tag{13}$$

where $\rho_{\mathbf{z}}(\mathbf{z})$ and $F_{\mathbf{z}}(\mathbf{z})$ are the Probability Density Function (PDF) and the Cumulative Distribution Function (CDF) of the random vector $\mathbf{Z}$.

In this work we will focus exclusively on non-intrusive methods, which require a minimal development effort. In particular the existing solvers are considered as black boxes and the non-intrusive methods need only to be wrapped around them. On the contrary, intrusive methods require the development of new solvers based on mixed discretization of the stochastic and the spatial models. These methods are usually better in dynamically adapting to time-dependent problems [7,6,4,32,29] but their implementation is often very demanding – sometimes prohibitive – for existing customized non-linear solvers.

3.1 Pseudo-random sampling methods

Among the existent techniques for Uncertainty Quantification, the random sampling methods are the most widely used. The most notable of these techniques, is the Monte Carlo method, developed in the late 40s. It is based

on the law of large numbers and it states that given the random vector $\mathbf{u} : \bar{D} \times (0, T] \times \mathbb{R}^d \to \mathbb{R}^m$ and the functional $g : \mathbb{R}^m \to \mathbb{R}^m$,

$$\frac{1}{n}\sum_{i=1}^{n} g\left(\mathbf{u}(\mathbf{x}, t, \mathbf{z}^{(i)})\right) \xrightarrow{a.s.} \mathbf{E}\left[g\left(\mathbf{u}(\mathbf{x}, t, \mathbf{Z})\right)\right] = \int_{\mathbb{R}^d} g\left(\mathbf{u}(\mathbf{x}, t, \mathbf{z})\right) dF_{\mathbf{z}}(\mathbf{z}), \quad (14)$$

for $n \to \infty$. In the definition above $\left\{\mathbf{z}^{(i)}\right\}_{i=1}^{n}$ is a ensemble of samples drawn from the probability distribution of $\mathbf{Z}$ and a.s. stands for *almost surely* implying convergence in probability. The implication of this property is that the realizations are always assumed meaningful. For $g_1 : \mathbf{u} \mapsto \mathbf{u}$ and $g_2 : \mathbf{u} \mapsto (\mathbf{u} - \mu_{\mathbf{u}})^2$,

$$\mathbf{E}\left[\mathbf{u}(\mathbf{x}, t, \mathbf{Z})\right] = \mathbf{E}\left[g_1\left(\mathbf{u}(\mathbf{x}, t, \mathbf{Z})\right)\right] \approx \frac{1}{n}\sum_{i=1}^{n} g_1\left(\mathbf{u}(\mathbf{x}, t, \mathbf{z}^{(i)})\right) = \bar{\mu}_{\mathbf{u}}(\mathbf{x}, t), \quad (15)$$

$$\mathbf{Var}\left[\mathbf{u}(\mathbf{x}, t, \mathbf{Z})\right] = \mathbf{E}\left[g_2\left(\mathbf{u}(\mathbf{x}, t, \mathbf{Z})\right)\right] \approx \frac{1}{n}\sum_{i=1}^{n} g_2\left(\mathbf{u}(\mathbf{x}, t, \mathbf{z}^{(i)})\right) = \bar{\sigma}^2_{\mathbf{u}}(\mathbf{x}, t). \quad (16)$$

The probabilistic error of these approximations is reduced asymptotically as $\mathcal{O}\left(1/\sqrt{n}\right)$ for number of realizations growing, i.e. $n \to \infty$. In spite of this slow convergence rate, Monte Carlo methods are widely used [27] due to their robustness, ease of use and to the fact that they do not suffer the *curse of dimensionality*, i.e., their convergence rate is independent from d. Due to all these properties, MC method is useful to generate reference solutions and for comparison with other techniques, but not for intractable problems that require significant effort. Thus, with the present technology, MC method cannot be used for all problems despite its robustness.

The slow convergence rate means that the ensemble size needed to resemble the target distribution must be large. Improvements of the Monte Carlo method have been proposed in order to cover more uniformly the stochastic domain, obtaining improved convergence rates, not always in the worst case scenarios, but in the average scenarios.

One of these methods is the Latin Hyper Cube method (LHC) [25], where the stochastic domain is divided in n equiprobable bins along each dimension and samples are taken such that each bin contains only one sample. This produces an ensemble that covers more uniformly the stochastic space and provides a better convergence rate in the average cases, even if the worst case convergence rate remains $\mathcal{O}\left(1/\sqrt{n}\right)$. A drawback of LHC is that the sample size need to be known a priori, and thus it is not suitable for incremental sampling.

An other notable method is the Quasi Monte Carlo (QMC) [26], where low discrepancy sequences of points are generated such that the domain is uniformly covered. The convergence rate of QMC is improved to $\mathcal{O}(\ln(n)^d/n)$, but the stochastic dimensionality of the problem becomes important.

3.2 Deterministic sampling methods

In the following we will handle functions with finite variance, i.e. belonging to the weighted $L^2_{\rho_{\mathbf{z}}}$ space defined as

$$L^2_{\rho_{\mathbf{z}}} = \left\{ f : \mathbb{R}^d \to \mathbb{R} \,\middle|\, \int_{\mathbb{R}^d} f^2(\mathbf{z})\rho_{\mathbf{z}} d\mathbf{z} = \mathbf{Var}[f(\mathbf{Z})] < \infty \right\}, \tag{17}$$

with inner product and norm defined as

$$(f,g)_{\rho_{\mathbf{z}}} = \int_{\mathbb{R}^d} f(\mathbf{z})g(\mathbf{z})\rho_{\mathbf{z}} d\mathbf{z}\,, \qquad \|f\|_{\rho_{\mathbf{z}}} = \sqrt{(f,f)_{\rho_{\mathbf{z}}}}\,. \tag{18}$$

For many standard distributions with density $\rho_{\mathbf{z}}$, we can find $\{\Phi_i(\mathbf{z})\}_{i=0}^N \subset \mathbb{P}^N$ that form a basis for $L^2_{\rho_{\mathbf{z}}}$ [38,37]. If the distribution is not standard, but has a density, then one can still use Gram-Schmidt orthogonalization to create suitable polynomials (see [16,15]). We can then define a projection operator P_N from $L^2_{\rho_{\mathbf{z}}}$ onto $\operatorname{span}\{\Phi_i(\mathbf{z})\}_{i=0}^N$ as

$$f(\mathbf{z}) \approx \tilde{f}(\mathbf{z}) = P_N f(\mathbf{z}) = \sum_{i=0}^{N} \hat{f}_i \Phi_i(\mathbf{z}), \qquad \hat{f}_i = \frac{(f,\Phi_i)_{\rho_{\mathbf{z}}}}{\|\Phi_i\|_{\rho_{\mathbf{z}}}}. \tag{19}$$

This provides an approximation $\tilde{f}$ of the target function f that is known as the *generalized Polynomial Chaos (gPC) expansion* of f. This gPC-expansion can be thought as a *surrogate function* of f. The computation of statistics from such surrogate function can be done easily. For example,

$$\mathbf{E}[f(\mathbf{z})] \approx \mathbf{E}[\tilde{f}(\mathbf{z})] = \hat{f}_0, \tag{20}$$

$$\mathbf{Var}[f(\mathbf{z})] \approx \mathbf{Var}[\tilde{f}(\mathbf{z})] = \sum_{i=1}^{N} \hat{f}_i^2 \left\|\Phi_i\right\|^2_{\rho_{\mathbf{z}}}, \tag{21}$$

where orthogonality of the basis $\{\Phi_i(\mathbf{z})\}_{i=0}^N$ is exploited.

The convergence of the polynomial approximation in (19) is spectral (super linear) if f is a smooth function and otherwise algebraic, cf. [16,5]. In order to obtain the surrogate model in (19), we are left with the computation of the coefficients $\hat{f}_i$ in (19). These can be obtained by means of two methods: the *Galerkin method*, where a reformulation of (8) in terms of stochastic modes is required, or the *Collocation method*, where approximations of $\hat{f}_i$'s are obtained by solving (8) on carefully selected points in the stochastic space. The Galerkin method is *intrusive*, i.e. the problem needs to be reformulated, thus it is cumbersome to be carried out for complex systems and will not be covered in this work (see [37,24] for an introduction to stochastic Galerkin methods). On the contrary the collocation method is *non-intrusive* and thus any existing deterministic solver for (1) can be used without modification.

3.2.1 Stochastic Collocation Method

The idea of the stochastic collocation method is to produce an ensemble of solutions $\mathbf{u}^{(j)}$, $i = 1, ..., M$ obtained by deterministically solving the governing equations (8) subject to carefully selected choices of M parameters $S_N = \{\mathbf{z}^{(j)}\}_{j=1}^{M}$ in the stochastic domain, in order to enable high accuracy in the evaluation of the coefficients of the gPC-expansion (19). An alternative approach is the interpolation method, but this is out of the scope of this work (see [37,24]). In order to simplify the notation in the following, we start considering functions of one stochastic parameter in the $L^2_{\rho_z}$ space (17), i.e. $d = 1$.

A set of orthogonal polynomials $\{\phi_i(z)\}_{i=0}^{N}$ that form a basis for $L^2_{\rho_z}$ can be found as explained in the preceding section. The expansion coefficients in (19) can then be found approximately as

$$\hat{u}_i = \frac{(u, \phi_i)_{\rho_z}}{\|\phi_i\|_{\rho_z}} = \frac{\int_{\mathbb{R}} u(z)\phi_i(z)\rho_z(z)dz}{\int_{\mathbb{R}} \phi_i^2(z)\rho_z(z)dz} \approx \frac{\sum_{j=0}^{M} u\left(z^{(j)}\right)\phi_i\left(z^{(j)}\right)w^{(j)}}{\sum_{j=0}^{M}\phi_i^2\left(z^{(j)}\right)w^{(j)}}, \quad (22)$$

where $\left\{\left(z^{(j)}, w^{(j)}\right)\right\}_{j=0}^{M}$ are Gauss-type quadrature points that can be readily obtained using the Golub-Welsch method [18]. These rules are exact when the integrand have a polynomial order up to $2M + 1$. The method is thus fully non-intrusive, since only deterministic solutions at particular points of the stochastic space are needed. This procedure differs from the classical Monte Carlo method only by the sampling technique used to select collocation nodes in the associated stochastic space.

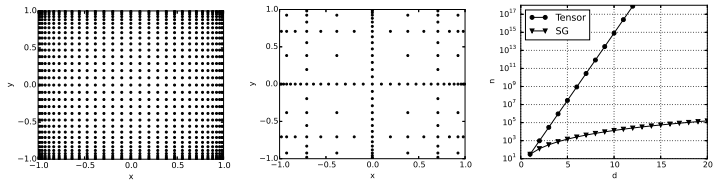
Let now $\mathbf{Z} : \Omega \to \mathbb{R}^d$ be a vector of independent random variables $Z_1, \ldots, Z_d$ with densities $\rho_{z_1}, \ldots, \rho_{z_d}$. It holds that $\rho_{\mathbf{z}}(\mathbf{z}) = \prod_{i=1}^{d} \rho_{z_i}(z_i)$, due to the independence condition. A possible set of basis functions for $L^2_{\rho_{\mathbf{z}}}$ is given by $\{\Phi_{\mathbf{i}}\}_{\max \mathbf{i} \leq P}$ where $\mathbf{i} = (i_1, \ldots, i_d)$ is a multi-index and

$$\Phi_{\mathbf{i}}(\mathbf{z}) = \phi_{i_1}(z_1) \cdot \ldots \cdot \phi_{i_d}(z_d). \quad (23)$$

This construction includes P^d basis functions and is more accurate than the required order P. An alternative set of basis is given by $\{\Phi_{\mathbf{i}}\}_{|\mathbf{i}|=0}^{P}$, where $|\mathbf{i}| = \sum_{j=1}^{d} i_j$. For this set of basis

$$\dim\left(\operatorname{span}\{\Phi_{\mathbf{i}}\}_{|\mathbf{i}|=0}^{P}\right) = \binom{N+d}{N}, \quad (24)$$

that is more tractable than P^d. The computation of the coefficients in the multidimensional gPC-expansion is again possible using Gauss-type quadrature rules. Both of these constructions are based on the tensor product of 1-dimensional rules and thus are computationally expensive: for a 1-dimensional quadrature rule using M points, the d-dimensional cubature rule uses M^d points. This effect is called *curse of dimensionality*.



(a) Fejér's tensor grid (b) Fejér's sparse grid (c) Fejér's scaling with d

Fig. 2: A tensor grid and a sparse grid of order $l = 5$ based on Fejér's quadrature rule [14,33]. On the right it is illustrated how the number of quadrature points scale with respect to d for $l = 5$.

Before addressing the curse of dimensionality, we need first to observe that quadrature rules based on the zeros of orthogonal polynomials are not *nested* in general, meaning that the quadrature points Θ_l based on the polynomial of order l are not in $\Theta_{l'}$, with $l' \geq l$. This property is important in practical calculations in case we would like to increase the accuracy but we don't want to waste results already computed. Common choices of nested quadrature rules are the Clenshaw-Curtis and Fejér's [8,14,33], that uses the maxima of the Chebyshev polynomials as quadrature points, and the Kronrod-Patterson rules [20]. With appropriate scaling, this quadrature rule works on the bounded interval $[0, 1]$ with a probability density function $\rho(z) = 1$, corresponding to the uniform distribution. In general we will have to compute integrals as in (22), where ρ_z does not need to be the uniform density function. However, using the fact that the CDF F_z is bijective, we can use a variable transformation s.t.

$$\int_{\mathbb{R}} f(z)\rho_z(z)dz = \int_0^1 g(x)dx, \qquad g(x) \equiv f(F_z^{-1}(x)). \tag{25}$$

Using these nested rules, we can attempt to alleviate the curse of dimensionality. One particularly successful approach is given by *Sparse Grids* (see [28,9] for details). The idea is not to take the complete tensor product of the 1-dimensional grids, but only products up to the desired order for each stochastic dimension, very much alike the construction of the set of basis $\{\Phi_{\mathbf{i}}\}_{|\mathbf{i}|=0}^{P}$. This procedure assumes a certain level of separability of the function, meaning that the cross-contribution of the parameters is lower than the contribution of the parameters considered separately. Figure 2 shows a comparison of tensor grids and sparse grids. From figure 2c we can see that the gain given by sparse grids over tensor grids increases with the stochastic dimension d. Using sparse grids the curse of dimensionality is alleviated, even if good accuracy in high dimensions is still a demanding task.

4 Uncertainty Quantification in Nonlinear Water Wave Simulations

We now use the stochastic formulation given in (8) to describe the stochastic evolution of water waves. We seek the stochastic free surface position $\zeta(\mathbf{x}, t, \mathbf{Z})$ and the stochastic velocity potential $\tilde{\phi}(\mathbf{x}, t, \mathbf{Z})$. For both the Monte Carlo approach and the Stochastic Collocation method, we need to solve (8) at a set of points $\left\{\mathbf{z}^{(i)}\right\}_{i=1}^{N}$, producing the ensemble of solutions

$$\left\{\zeta\left(\mathbf{x}, t, \mathbf{z}^{(i)}\right)\right\}_{i=1}^{N}, \text{ and } \left\{\tilde{\phi}\left(\mathbf{x}, t, \mathbf{z}^{(i)}\right)\right\}_{i=1}^{N}. \tag{26}$$

The sampling strategy depends on the particular method chosen. Furthermore, the Stochastic Collocation Method constructs surrogate functions

$$\zeta(\mathbf{x}, t, \mathbf{Z}) \approx P_N \zeta(\mathbf{x}, t, \mathbf{Z}) = \sum_{|\mathbf{i}| \leq N} \hat{\zeta}_{\mathbf{i}}(\mathbf{x}, t) \Phi_{\mathbf{i}}(\mathbf{Z}), \tag{27a}$$

$$\tilde{\phi}(\mathbf{x}, t, \mathbf{Z}) \approx P_N \tilde{\phi}(\mathbf{x}, t, \mathbf{Z}) = \sum_{|\mathbf{i}| \leq N} \hat{\tilde{\phi}}_{\mathbf{i}}(\mathbf{x}, t) \Phi_{\mathbf{i}}(\mathbf{Z}), \tag{27b}$$

that provide an easy way to compute statistics and to reproduce the PDFs of the solution variables.

In the following, we revisit two classical benchmarks to illustrate how uncertainty quantification can be done efficiently. Even if both Monte Carlo method and Stochastic Collocation method have been employed, due to space constraint, only the figures obtained using SCM will be shown. In all the cases presented the results agree for the two methods and SCM shows faster convergence and thus requires much fewer realizations, resulting in reduced CPU time.

4.1 Harmonic generation over a submerged bar

We now consider an experimental setting originally proposed by Beji and Battjes (1994) [1]. In the experiment a nonlinear wave propagates across a submerged bar. In the process the bound wave harmonics are decomposed into free harmonics which are released on the lee side of the bar and causes a transformation of the initial wave profile as described in [2]. It is generally accepted that the experiment can be reproduced within engineering accuracy by a verified deterministic wave model such as (1), which describe both the nonlinear and dispersive effects accurately. However, calibration details and measurement errors are not included in the public report by Beji and Battjes. Therefore, in the following, we will assume uncertainties and detail how these can be accounted for in the stochastic simulations.

To analyze the wave evolution we use the bottom topography of the experiments shown in figure 1. We consider the setup corresponding to Case A in the original experiments [2], where the input wave signal is defined by a wave

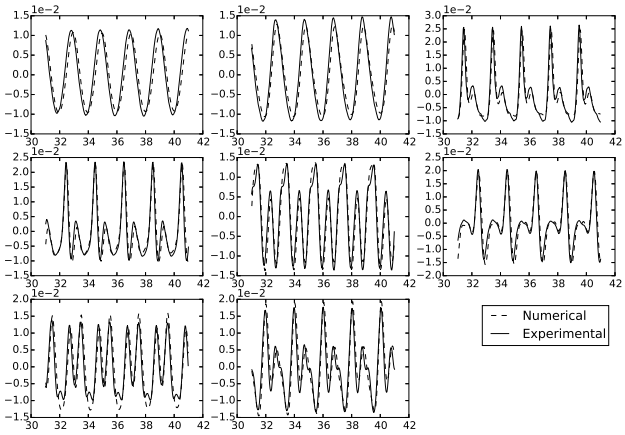


Fig. 3: Deterministic solution of the submerged bar experiment at eight different gauge locations. The experimental data are due to Luth *et al.* [23].

period $T = 2.02$s and a wave height $H = 2$cm. In the numerical solver the input waves are generated using Stokes second order theory.

The shape of the bar and the shape of the incoming wave influence the spectrum of the waves that reach the right end of the domain as analyzed in [2]. In the following different sources of uncertainties are considered and the results are compared with deterministic results often presented in existing literature as well as to the experimental measurements due to Luth *et al.* [23].

4.1.1 Deterministic results

As a conventional mean for validation of the numerical wave model, we compare with the experimental measurements at eight gauges positioned in the wave tank. The results of this comparison are presented in Figure 3, where the bathymetry used is the exact bathymetry illustrated in figure 1. The results have been computed with the parameters of the experiment given in table 1. These parameters will be changed in the following to reflect single realizations of uncertain parameters. Clearly, the computation and the experiments match qualitatively very well, however there are noticeable discrepancies between the numerical calculations and the experimental data. For example, the wave heights and phases are not exactly reproduced at the first and second measurement stations. Discrepancies in the wave signal are observed at the high peaks in measurements from stations 5, 7 and 9, and in the low and intermediate peaks of station 6. The numerical calculations are done using

Description	Variable	Value
Bar height from bottom	h_{bar}	$0.3m$
Bottom floor	h_b	$-0.4m$
Entering wave length	T	$2.02m$
Basin Length		$29.0m$
Gauges positions		$\{4.0, 10.5, 13.5, 14.5, 15.7, 17.3, 19.0, 21.0\}$

Table 1: Nominal values and experimental settings used for the deterministic solution of the water wave problem.

a high-order accurate numerical method [11] with sufficient resolution to accurately resolve the dispersion and nonlinear wave effects, and are therefore assumed to be converged to a grid-independent solution. The absorption zone introduced behind the bar has been defined so that minimum wave reflections occur. However, discrepancies between experiments that are due to uncertainties in the measurement data or experimental setup are not taken into account in the numerical results. This motivates the studies in the following, where we will investigate the effects of taking into account the uncertainty in the model input.

4.1.2 Uncertain still water height

A very difficult parameter to be controlled when experiments in a manufactured basin are performed, is the exact height of the still water. In particular the accuracy of the measured height is sensitive to fluctuations, evaporation and spill of water. Here we use the truncated normal distribution

$$h_b \sim \text{tr}\mathcal{N}(0.3m, 0.0125^2 m^2, [0.375m, 0.425m]) \tag{28}$$

to represent the fact that large defects in the water height can be detected and corrected.

Figure 4 shows the mean and standard deviation of the solution computed with SCM with 11 realizations. We can notice that the mean value seem not to follow the experimental solution on top and downstream of the bar. To shed more light on the characteristics of the distribution of the solution, we need to look at its PDF. We can use the surrogate model (19) of the solution, obtained by the SCM, to evaluate a high number of approximate realizations of the model with insignificant computational burden. We generate in this way 10^4 realizations sampled from the distribution (28). The surrogate solutions are then organized in histograms and which are shown in figure 5. In spite of being generated by a Gaussian input uncertainty, the distribution of the solution is not Gaussian and higher statistical moments have developed during the propagation of such uncertainties. In particular we can see that at certain times, the distribution is even bimodal due to an uncertain phase shift.

The quality of the surrogate model (19) can be checked inspecting the decay of the expansion coefficients (19). If the solutions change smoothly in the parameter space (h_b), then the convergence is expected to be very fast,

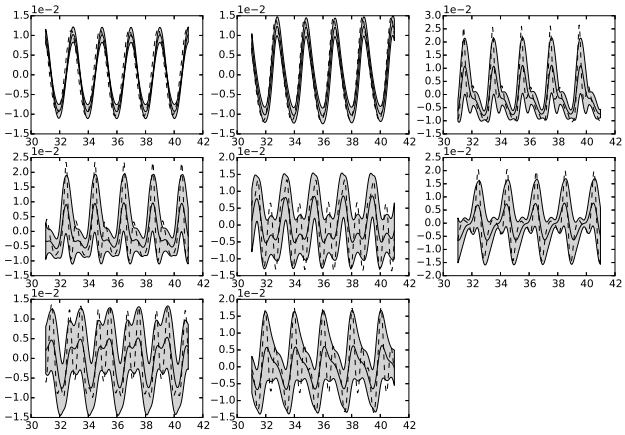


Fig. 4: Mean (solid line) and standard deviation (shaded) of the solution of the submerged bar experiment with $h_b \sim \mathrm{tr}\mathcal{N}(0.3m, 0.0125^2m^2, [0.275m, 0.325m])$ at the different measurement locations, obtained by the SCM with 11 realizations. The experimental data (dashed) is also shown.

in the best case *spectral* [5]. Figure 6 shows the time varying values of the coefficients in the $\log_{10}$ scale. We can notice a clear periodicity in the values of the coefficients. This highlights that the expansion order needed to reach a required accuracy may vary over time and care should be taken to select an appropriate expansion order for the analysis to be accurate.

The convergence rates of the MC and the SCM methods are shown in Fig. 7a. Here, for each gauge, the L^2 error in the estimated time-varying mean is computed against an highly accurate reference solution obtained using SCM with polynomial order 20. The convergence is shown in terms of number of function evaluations. Assuming that the computational complexity of the deterministic problem is mildly dependent on the examined parameters, the number of function evaluations is linear to the CPU time required. We can recognize the convergences on the first two gauges which are significantly faster. The MC method shows its characteristic slow convergence, while the SCM exhibit *spectral* convergence.

4.1.3 Uncertain input wave length

A parameter which is difficult to reproduce accurately in practical experiments is the input wave signal. An accurate representation of the wave signal requires that the wave maker displacement and the wave amplitudes are matched. This

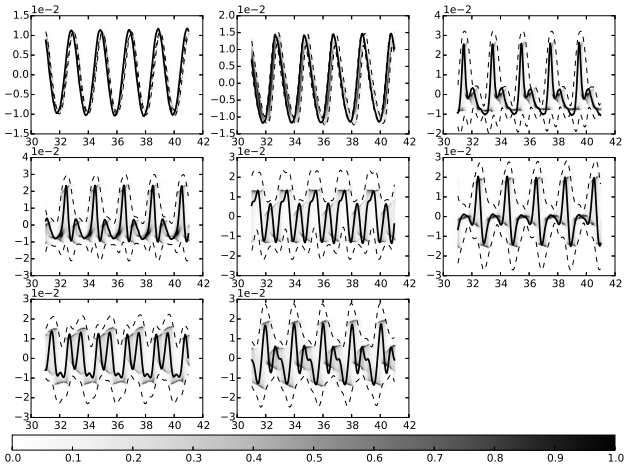


Fig. 5: Probability distributions of the time-varying solution of the submerged bar experiment with $h_b \sim \mathrm{tr}\mathcal{N}(0.3, 0.0125^2, [0.275, 0.325])$ m at different measurement locations. These results are obtained by the SCM with 11 realizations. The thick black lines show the experimental results at the different gauges, while the dashed lines show the 95% confidence intervals.

can be difficult to achieve in practice, especially for nonlinear wave signals, and may lead to harmonic generation. To illustrate how such uncertainty in the signal can be accounted for, we use

$$T \sim \mathcal{N}(2.02s, 0.01^2 s^2) \tag{29}$$

to represent the uncertainty due to the generation of the input waves.

Using the surrogate model provided by (19), we can reproduce the time dependent probability distribution of the solution shown in figure 8. We can observe that the uncertainty on the input wave length gives a relatively small contribution to the uncertainty of the solution. A comparison of the convergence of the MC method and the SCM method is shown in Fig. 7b.

4.1.4 Two dimensional uncertainty on wave length and water height

In many practical cases uncertainty does not enter a dynamical system only through one coefficient, but as a combination of multiple uncertainties. The still water height and the input wave length enter the system independently, however they will have a combined influence on the uncertainty of the solution. We will use the distributions (28) and (29) as in the previous examples. We

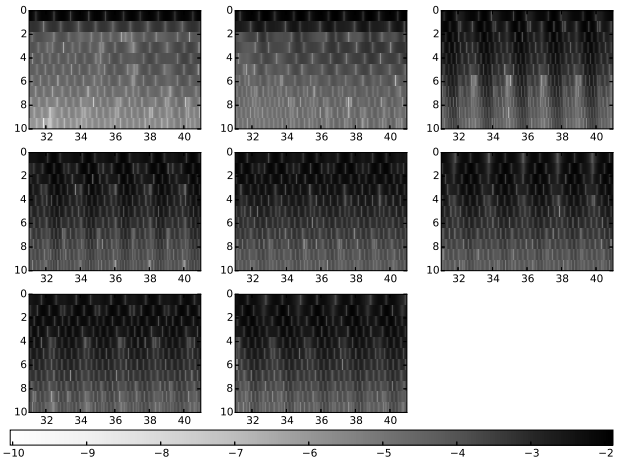
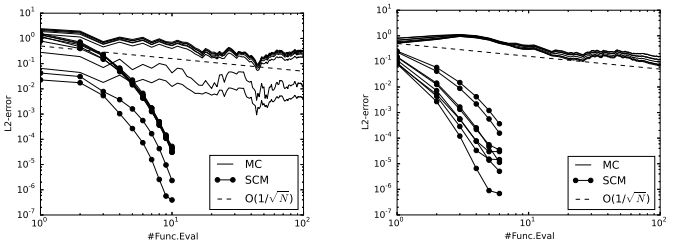


Fig. 6: Decay of the gPC-expansion coefficients in (19). The time varying coefficients values are shown in the $\log_{10}$ scale according to the colors in the color bar.

expect the variance of the solution to increase due to the higher quantity of uncertainty allowed in the system. Figure 9 shows the time dependent distribution function of the solution. We can observe that the obtained uncertainty is not merely the superposition of the uncertainties obtained in the one dimensional cases (see fig. 5 and fig. 8), but is increased due to the interaction between the two. This effect will be more evident through the observation of the coefficients in the gPC expansion (19).

High order cubature rules of order 20 were used in the gPC method. This expansion order was found sufficient to get enough accuracy in the construction of the surrogate function (19). Figure 10 shows the decay of the projection coefficients in relation to both the input uncertainties. A total independence of the two parameters in the influence of the system would produce an expansion (19) where all the non-zero coefficients $\hat{f}_{\mathbf{i}}$ are the ones with $\mathbf{i} = (i, 0)$ or $\mathbf{i} = (0, j), i, j = 0, \ldots, 20$. This corresponds to have decays similar to the one shown in the first upper-left plot in figure 10. The next plots, however, show that the two input uncertainties act on the solution in non-trivial ways when combined. This means that the results of the UQ analysis on the two separate sources, cannot be trivially superposed, but they need to be considered together in a unique UQ analysis. Furthermore, the application of Sparse Grids on this case would suffer from this property which corresponds to the lack of separability



(a) Uncertain still water height (b) Uncertain wave length

Fig. 7: Convergence rate of the MC method and the SCM method. The L^2 error of the approximation of 10s of simulation is computed against an highly accurate reference solution of order 20. The different lines belong to different gauges. The MC method exhibit its slow convergence of $\mathcal{O}(1/\sqrt{N})$. The SCM method shows *spectral* convergence.

of the function of interest. Methods which allow a sparse sampling in these situations are still lacking in the scientific literature.

4.1.5 Uncertain bottom topography

The topography of the basin is often precise in experimental settings, but rarely for real sites. Small discrepancies with respect to the ideal design can still be present. We will model these discrepancies using a Gaussian random field added on top of the deterministic basin, as shown in figure 1. In particular we will consider a Gaussian random field with exponential covariance (11) and with correlation length $a = 1.0$. The mean of the field is set to be the nominal bottom topography and the total variance of the field is set to $\sigma^2 = 0.01^2$. One realization of such random field is shown in figure 1a. With this model we try to capture small macroscopic errors in the slope of the basin's bottom. The random field is expanded using the KL-expansion (9), capturing 95% of the total variance of the field. This results in a truncated KL-expansion with 3 terms.

The Sparse Grid method introduced in Section 3.2.1 is used here with order $l = 3$ to compute mean and variance of the free surface profile at the eight measuring stations. Figure 11 shows the results obtained using only 19 realizations of the deterministic model. We can see that the uncertain bottom topography considered plays an important role in the wave transformation downstream of the bar, even if the random field considered has a relatively long correlation length and small variance.

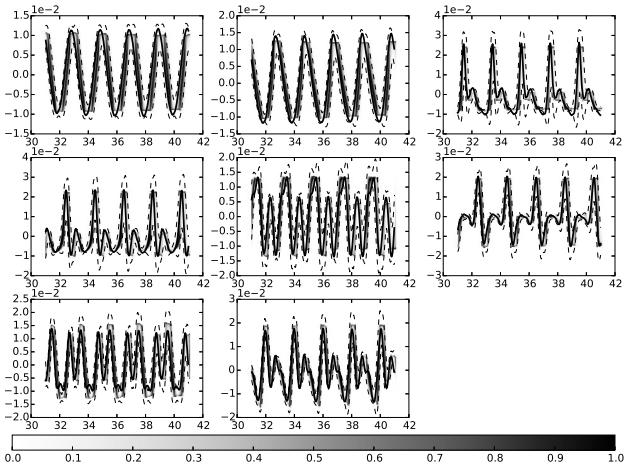


Fig. 8: Probability distributions of the time-varying solution of the submerged bar experiment with input wave length $T \sim \mathcal{N}(2.02s, 0.01^2 s^2)$ at different measurement locations. The thick black lines show the experimental results at the different gauges, while the dashed lines show the 95% confidence intervals.

4.2 Harmonic generation over a semi-circular shoal

Extending the analysis to the full three dimensional problem we will proceed to the experiments of Whalin [34]. The experiments consists of a regular wave propagating over a semi-circular shoal, see figure 12a. The shoaling process transfer energy between the bound harmonics but, in contrast to the submerged bar case, the harmonics remain bounded and refraction adds complexity to the solution. The Whalin experiments have become standard benchmarks for dispersive wave models regardless of a rather substantial scatter present in the experimental data. We will look into the case of of incoming waves with height $H = 0.015$ m and period $T = 2$ s. For this case most numerical models tend to over predict the amplitude of the second harmonic. As the present model is able to accurately capture all the major phenomena taking place in the experiments we are interested to see what level of uncertainty this corresponds to in the experimental values.

The deterministic numerical solution is computed for $t \in [0, 50]$ and then compared with the experimental measurements of the magnitude of the first three harmonics at different measurement locations through the center plane. Figure 12b shows the fitting of the numerical solution to the measurement data. The aim of the next sections is to study how the uncertainty in some

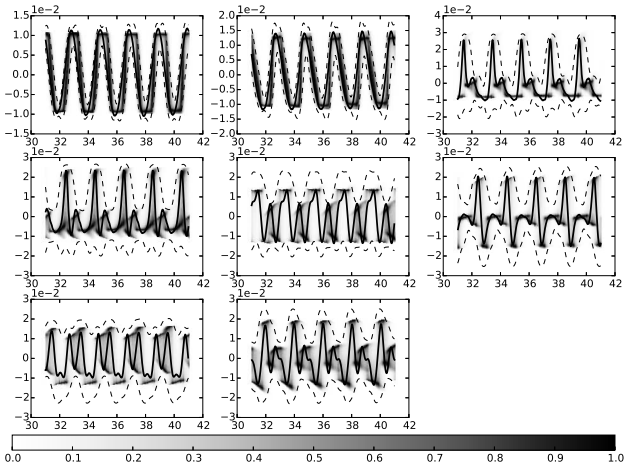


Fig. 9: Probability distributions of the time-varying solution of the submerged bar experiment with with uncertain still water height h_b and wave length T, at different measurement locations. The thick black lines show the experimental results at the different gauges, while the dashed lines show the 95% confidence intervals.

experimental parameters can influence the results. Without presumption of causality, this analysis can highlight parameters that can influence results more deeply than others. The computational cost of solving the full three dimensional problem calls for efficient UQ methodologies that require the minimum number of simulations to make analysis practically feasible.

4.2.1 One dimensional uncertainties

Building up on the experience acquired on the two dimensional case and from experimental knowledge, we will focus our attention to the two parameters that are most difficult to match, namely the input wave period and height. Due to the lack of information about how accurate experiments can be, we will assume that the input parameters are described by a Gaussian distribution and we will try to evaluate how sensitive the system is to single uncertainties, and, in the next section, to the combination of the two. We will model the wave height and the wave period with Gaussian distributions centered on their nominal values and with 5% standard deviation.

A stochastic collocation approach with estimation of the generalized Polynomial Chaos expansion (19) is adopted, with the order dictated by the accuracy required. Figure 13 shows the mean and the 95% confidence interval as

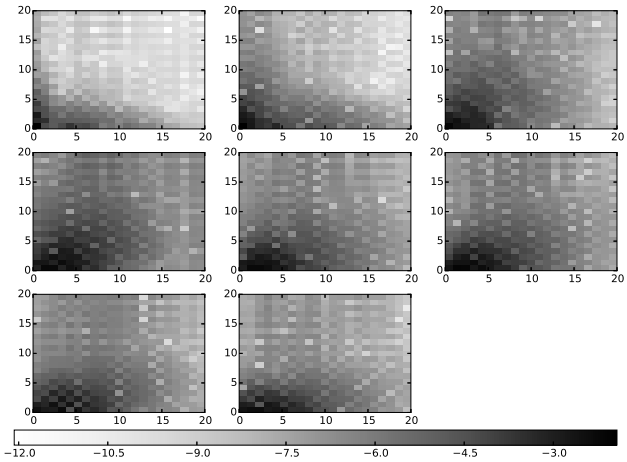


Fig. 10: Decay of the 2-dimensional gPC-expansion coefficients in (19) for the last integration time at different measurement locations. The coefficient values are shown in the $\log_{10}$ scale according to the colors in the color bar.

well as the space-dependent distribution of the harmonics and the fitting with the experimental data.

4.3 Two dimensional uncertainty

The same problem setting is now investigated with uncertainty on the wave height and period at the same time. The same distributions used in the one dimensional setting are used here for the uncertainty sources. The Stochastic Collocation Method of order 5 is used to compute the space dependent probability distribution of the first three harmonics of the propagated wave. A total of only 36 deterministic simulations are required to obtain the desired approximation.

Figures 14 shows the space dependent mean and 95% confidence interval of the first three harmonics, as well as their space dependent probability distribution. Again we can notice that the resulting uncertainty – measured in variance of the solution – is not the mere superposition of the variances obtained with one dimensional uncertainties (see fig. 13). The probability distribution of the first three harmonics seem now to include the experimental measurements within some high probability region.

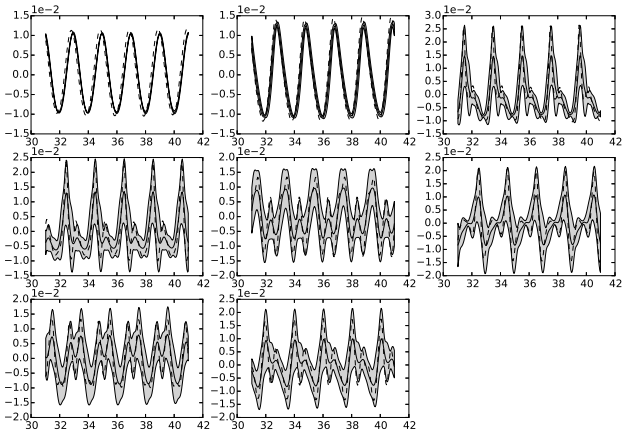
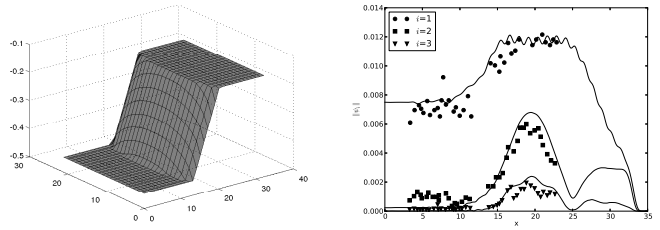
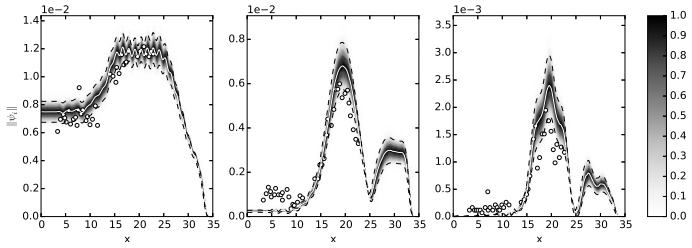


Fig. 11: Mean (solid line) and standard deviation (shaded) of the solution of the submerged bar experiment with the bottom topography described by a Gaussian random field with Ornstein-Uhlenbeck [31] covariance function at the different measurement locations. The experimental data (dashed line) is also shown. The results are obtained using the Sparse Grid method with 19 function evaluations.

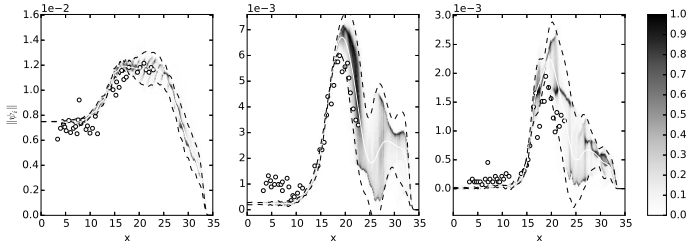


(a) Bottom Topography (b) Deterministic Solution

Fig. 12: Deterministic solution of the wave propagation in three dimensions. The first three harmonics of the numerical solution (full lines) for the centerline are compared with the corresponding experimental measurements at different longitudinal locations in the basin (dots).



(a) Wave Height



(b) Wave Period

Fig. 13: Reconstructed space-dependent probability distribution function of the three harmonics of the solution of the Whalin test with one dimensional uncertainty. The white line shows the space-dependent mean, while the dashed lines show the 95% confidence interval around the mean. The scattered dots show the experimental data results.

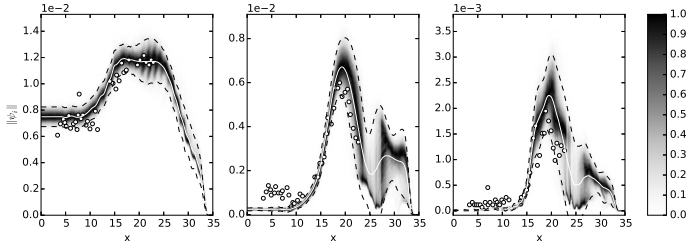


Fig. 14: Space-dependent probability distribution function of the Whalin test with two-dimensional uncertainty. The white solid line represent the mean for the three harmonics. The dashed lines show the 95% confidence interval around the mean. The scattered dots are the experimental measurements.

5 Conclusions

The Stochastic Collocation method (SCM) for Uncertainty Quantification (UQ) has been applied to a variety of stochastic wave problems, where the stochastic parameter space is low-dimensional. Numerical experiments have been carried out adopting often used standard benchmarks for wave models to create new stochastic benchmarks. The type of uncertainties accounted are those that are likely to appear in experimental settings, such as the input wave characteristics, the water height, and the topography of manufactured basins. With the aim of constructing stochastic benchmarks, we made reasonable assumptions regarding the distribution of such uncertainties, that would otherwise need to be characterized by extensive measurements or by a better description of the believed distributions. The focus in this work is toward the approximation of the probability distributions of observable Quantities of Interest (QoIs) and the exploration of the available methods able to reach this goal with the lowest computational burden.

The UQ methods selected here are all designed to tackle types of problems where obtaining a single deterministic solution is computationally expensive. The SCM, in its tensor form or in a sparse grid form, allows for high accuracy, due to the possibility of spectral convergence of the approximation, using a small number of deterministic solutions of the problem, that are obtained in a non-intrusive way. This allows the adoption of existing solvers, whose source code might be complex or not even available. In this work we use a high-order finite difference method for a fully nonlinear and dispersive medium-scale water wave model [11]. Thanks to the advent of High Performance Computing [12], large-scale water wave simulations are becoming more and more efficient and are subject of ongoing research.

On the downside of the SCM there is the fact that it is only suitable for problems with a low-dimensional stochastic space. Reminding that each problem has its own peculiarities (e.g. stochastic dimension, deterministic computational complexity, etc.), we are aware that as the stochastic dimension increases, the number of required solutions of the deterministic model increases more than polynomially, getting quickly not feasible. At the current state of research, these cases must be addressed using pseudo-random sampling techniques, whose convergence is slow but independent from the stochastic dimensionality. It is however important to remind that the latter techniques aim at the approximation of the output distribution through the approximation of its moments and this gets increasingly expensive if the target distribution has many non-zero high moments.

It is also important to observe that not all high-dimensional problems are really high-dimensional. One case was shown for the uncertain bottom topography in the submerged bar experiment, where the random field describing the perturbation of the topography has some regularity properties and can be parametrized via the KL-expansion, transforming an ideally infinite dimensional problem to a finite dimensional one.

The analysis performed on the new stochastic benchmarks show that the uncertainties on the input wave characteristics and the bottom topography have indeed a relevant effect on the free surface solutions. These effects are amplified when these uncertainties are considered simultaneously, leading to a non trivial transformation of the input probability distribution. The results of such analysis can be considered when explaining some of the discrepancy between numerical solutions and experimental results. In ongoing works we are considering problems with higher stochastic dimensions, resorting to novel techniques for deterministic sampling, such as Adaptive Sparse Grids [9] and Spectral Tensor Train decomposition [3].

The frameworks for random sampling[2] and for SCM[3], as well as the results obtained in this work[4], are made available on-line and are general enough to be applied on both small-scale and large-scale problems with no additional implementation burden.

References

1. Beji, S., Battjes, J.A.: Numerical simulation of nonlinear-wave propagation over a bar. Coastal Engineering **23**, 1–16 (1994)
2. Benxia, L., Xiping, Y.: Wave decomposition phenomenon and spectrum evolution over submerged bars. Acta Oceanologica Sinica **28**(3), 82–92 (2009)
3. Bigoni, D., Engsig-Karup, A.P., Marzouk, Y.M.: Spectral tensor-train decomposition (2014)
4. Boyaval, S., LeBris, C., Lelièvre, T., Maday, Y., Nguyen, N.C., Patera, a.T.: Reduced Basis Techniques for Stochastic Problems. Archives of Computational Methods in Engineering **17**(4), 435–454 (2010)
5. Canuto, C., Hussaini, M., Quarteroni, A., Zang, T.: Spectral Methods - Fundamentals in Single Domains. Scientific Computation. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
6. Cheng, M., Hou, T.Y., Zhang, Z.: A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations I: Derivation and algorithms. Journal of Computational Physics **242**, 843–868 (2013)
7. Cheng, M., Hou, T.Y., Zhang, Z.: A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations II: Adaptivity and generalizations. Journal of Computational Physics **242**, 753–776 (2013)
8. Clenshaw, C.W., Curtis, A.R.: A method for numerical integration on an automatic computer. Numerische Mathematik **2**(1), 197–205 (1960)
9. Conrad, P., Marzouk, Y.: Adaptive Smolyak pseudospectral approximations. SIAM Journal on Scientific Computing **35**(6), 2643–2670 (2013)
10. Dutykh, D., Clamond, D.: Efficient computation of steady solitary gravity waves (2013)
11. Engsig-Karup, A.P., Bingham, H.B., Lindberg, O.: An efficient flexible-order model for 3d nonlinear water waves. Journal of Computational Physics **228**, 2100–2118 (2008)
12. Engsig-Karup, A.P., Glimberg, L.S., Nielsen, A.S., Lindberg, O.: Fast hydrodynamics on heterogenous many-core hardware. In: R. Couturier (ed.) Designing Scientific Applications on GPUs, Lecture notes in computational science and engineering. CRC Press / Taylor & Francis Group (2013)
13. Engsig-Karup, A.P., Madsen, M.G., Glimberg, S.L.: A massively parallel gpu-accelerated model for analysis of fully nonlinear free surface waves. International Journal for Numerical Methods in Fluids **70**(1), 20–36 (2011)

[2] `https://pypi.python.org/pypi/UQToolbox/`

[3] `https://pypi.python.org/pypi/SpectralToolbox/`

[4] `http://www2.compute.dtu.dk/~apek/OceanWave3D/`

14. Fejér, L.: Mechanische Quadraturen mit positiven Cotesschen Zahlen. Math. Z. **37**, 287–309 (1933)
15. Gautschi, W.: Algorithm 726: ORTHPOL;a package of routines for generating orthogonal polynomials and Gauss-type quadrature rules. ACM Trans. Math. Softw. **20**(1), 21–62 (1994)
16. Gautschi, W.: Orthogonal Polynomials: Computation and Approximation. Numerical Mathematics and Scientific Computation. Oxford University Press (2004)
17. Glimberg, L.S., Engsig-Karup, A.P., Dammann, B., Nielsen, A.S.: Development of high-performance software components for emerging architectures. In: R. Couturier (ed.) Designing Scientific Applications on GPUs, Lecture notes in computational science and engineering. CRC Press / Taylor & Francis Group (2013)
18. Golub, G.H., Welsch, J.H.: Calculation of Gauss Quadrature Rules. Mathematics of Computation **23**(106), 221–230 (1969)
19. Kiureghian, A., Ditlevsen, O.: Aleatory or epistemic? Does it matter? Structural Safety (2009)
20. Kronrod, A.S.: Nodes and Weights of Quadrature Formulas. English transl. from Russian, Consultants Bureau **35**(597) (1965)
21. Larsen, J., Dancy, H.: Open boundaries in short wave simulations - a new approach. Coastal Engineering **7**, 285–297 (1983)
22. Loève, M.: Probability Theory, vols. I-II, 4 edn. Comprehensive Manuals of Surgical Specialties. Springer, New York (1978)
23. Luth, H.R., Klopman, B., Kitou, N.: Projects 13G: Kinematics of waves breaking partially on an offshore bar: LDV measurements for waves with and without a net onshore current. *Technical report H1573*, Delft Hydraulics (1994)
24. Maître, O.P.L., Knio, O.M.: Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics. Springer (2010)
25. Mckay, M., Beckman, R., Conover, W.: A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code. Technometrics **41**(1), 55–61 (2000)
26. Morokoff, W.J., Caflisch, R.E.: Quasi-Monte Carlo Integration. Journal of Computational Physics **122**(2), 218–230 (1995)
27. Naess, A., Moan, T.: Stochastic Dynamics of Marine Structures. Cambridge University Press, Cambridge (2012)
28. Petras, K.: Smolyak cubature of given polynomial degree with few nodes for increasing dimension. Numerische Mathematik **93**(4), 729–753 (2003)
29. Sapsis, T.P., Lermusiaux, P.F.: Dynamically orthogonal field equations for continuous stochastic dynamical systems. Physica D: Nonlinear Phenomena **238**(23-24), 2347–2360 (2009)
30. Schwab, C., Todor, R.A.: KarhunenLoève approximation of random fields by generalized fast multipole methods. Journal of Computational Physics **217**(1), 100–122 (2006)
31. Uhlenbeck, G., Ornstein, L.: On the theory of the Brownian motion. Physical review **36**(1905) (1930)
32. Venturi, D.: On proper orthogonal decomposition of randomly perturbed fields with applications to flow past a cylinder and natural convection over a horizontal plate. Journal of Fluid Mechanics **559**, 215 (2006)
33. Waldvogel, J.: Fast Construction of the Fejér and ClenshawCurtis Quadrature Rules. Bit Numerical Mathematics **46**(1), 195–202 (2006)
34. Whalin, R.W.: The limit of applicability of linear wave refraction theory in convergence zone. Tech. Rep. H-71-3, US Army Corps of Engineers (1971)
35. Wojtkiewicz, S.J., Eldred, M., Field, R.J., Urbina, A., Red-Horse, J.: A toolkit for uncertainty quantification in large computational engineering models. In: Proceedings of the 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference (2001)
36. Xiu, D.: Fast numerical methods for stochastic computations: a review. Communications in computational physics **5**(2), 242–272 (2009)
37. Xiu, D.: Numerical Methods for Stochastic Computations: A Spectral Method Approach. Princeton University Press (2010)
38. Xiu, D., Karniadakis, G.E.: The wiener–askey polynomial chaos for stochastic differential equations. SIAM J. Sci. Comput. **24**(2), 619–644 (2002)

SPECTRAL TENSOR-TRAIN DECOMPOSITION

DANIELE BIGONI[†], ALLAN P. ENGSIG-KARUP[†], AND YOUSSEF M. MARZOUK[‡]

Abstract. The approximation of high-dimensional functions with surrogates is of key importance for the advance of uncertainty quantification and inference. We propose the construction of surrogate functions using a novel spectral extension of tensor-train decomposition and we provide error estimates for surrogate functions based on projection and interpolation. To this end, we define the functional version of the tensor-train decomposition and we use its properties to prove that the differentiability properties of the target function are preserved by the decomposition. The linear scalability with respect to the dimension of tensor-train decomposition is coupled with the spectral convergence rate of polynomial approximation, obtaining a method that is accurate and addresses the *curse of dimensionality* at the same time. The tensor-train decomposition of high-dimensional functions is obtained by the sampling algorithm `TT-DMRG-cross`, leading to a number of function evaluations that increases linearly with the dimension. To assess the properties of the method, the spectral tensor-train decomposition is applied on the Genz functions up to dimension $d = 100$. A new set of Genz functions is proposed, for which the difficulty of the approximation does not decrease with the dimension. The new method is additionally tested on an ad-hoc functions with mixed Fourier modes and local features, in order to highlight strengths and weaknesses. Finally the method is used to construct an approximation of the elliptic equation with random input data, where the diffusivity is modeled by a log-normal random field. The software and examples presented in this work are available on-line[1].

Key words. Approximation theory, tensor-train decomposition, orthogonal polynomials, uncertainty quantification.

AMS subject classifications. 41A10, 41A63, 41A65, 46M05, 65D15

1. Introduction.

High-dimensional functions appear frequently in engineering applications, where quantities of interest (QoIs) may depend in non trivial ways on a large number of variables. Problems with ten or more variables become quickly intractable with traditional approximation methods. In this work we will address this problem, by extending the discrete tensor-train format [31] with the spectral theory for polynomial approximation.

Problems involving the approximation of high-dimensional functions can be found in the field of Uncertainty Quantification, where parametric stochastic partial differential equations need to be solved and the number of stochastic parameters can exceed the hundreds or even the thousands. In particular the usage of approximations is important when the high-dimensional function is computationally expensive and its evaluation is required at many points in the high-dimensional space.

For a high-dimensional function $f : [a, b]^d \to \mathbb{R}$, the traditional approximation approach is based on its projection onto the space spanned by the tensor product of basis functions $\{\phi_{i_j}(x_j)\}_{i_j=1}^{n_j} \subset L^2([a, b])$ for $j = [1, \dots, d]$, obtaining:

$$f \simeq \sum_{i_1}^{n_1} \cdots \sum_{i_d}^{n_d} c_{i_1,\dots,i_d} \left(\phi_{i_1} \otimes \cdots \otimes \phi_{i_d}\right). \tag{1.1}$$

This approach quickly becomes impractical as the dimension d increases, due to the exponential growth in the number of coefficients $c_{i_1,\dots,i_d}$ and the computational effort (i.e., number of function evaluations) required to compute them. This effect is known as the *curse of dimensionality*.

Attempts made in order to tackle the curse of dimensionality are all based on some assumptions on the separability of the high-dimensional function to effectively reduce the number of unknown coefficients. For example, one of the most successful methods is the adaptive pseudospectral function approximation based on Smolyak Sparse Grids [1, 34, 8, 9, 13]: instead of taking the full tensor product of the basis functions as in (1.1), we consider a subset of admissible indices [8, 9] – where

[†]Technical University of Denmark, Kgs. Lyngby, DK-2800 Denmark
[‡]Massachusetts Institute of Technology, Cambridge, MA 02139 USA
[1]https://pypi.python.org/pypi/TensorToolbox/

admissible means that they need to fulfill a mutual condition in order to make the Smolyak construction work – and project the function f onto the space spanned by the corresponding basis. The list of admissible indices can be increased to meet user defined accuracy requirements.

Other attempts use the concept of multiplicative and additive separability of a function, solving a minimization problem to find, for example, the approximation

$$f \simeq \sum_{i=1}^{r} \gamma_{i,1} \otimes \cdots \otimes \gamma_{i,d}, \tag{1.2}$$

where $\gamma_{i,1}, \ldots, \gamma_{i,d} : [a,b] \to \mathbb{R}$, for $i = 1, \ldots, r$. We will discuss this approach and its drawbacks in Section 2.1.

In this work we will use and extend the *tensor-train decomposition* (TT-decomposition) [31] for the approximation of high-dimensional functionals. Also the TT-decomposition is based on the concept of separability, but it builds its format in a hierarchical way, solving many of the drawbacks of other tensor decompositions. We will use classical polynomial approximation theory to extend the *discrete* TT-decomposition, in order to construct a low rank approximation of f. To do this, we will develop the *functional* counterpart of the tensor-train decomposition and highlight its convergence properties. In particular we will prove that this approach can be applied to a wide class of functions in L^2 that satisfy a particular regularity condition. For this class of functions the weak differentiability of the original function is preserved on the decomposed one, allowing us to apply the theory on polynomial approximation on the latter. This approach will exploit any smoothness property of the function f in order to improve the accuracy of the approximation and lower the computational burden to construct its approximation.

Since we are considering a setting where f is computationally expensive to evaluate, only a limited number of function evaluations is possible in reasonable time and thus we need to resort to a sampling method. We will construct the tensor-train approximation using the `TT-DMRG-cross` technique [35], where the function f is considered as a black-box method and it has to be evaluated only at relevant points in the parameter space.

The approach will be tested for the construction of surrogate functions of the Genz functions and ad-hoc functions with different decays of the Fourier coefficients, in order to observe their relation with the ranks of the decompositions.

The paper is organized as follows. In Section 2 we will review some of the definitions and the properties of several tensor decomposition formats, focusing in particular on the tensor-train decomposition. Section 3 is devoted to the review of concepts about the approximation of functions in Sobolev spaces. In Section 4 the spectral tensor-train decomposition is defined in a constructive way and the regularity properties of such decomposition are presented. Section 4.3.3 presents the practical implementation of the algorithm. In Section 5 numerical examples are presented.

2. Tensor decompositions. For the moment let us assume that we can afford the evaluation of the function $f : [a,b]^d \to \mathbb{R}$ at all points on a tensor grid $\boldsymbol{\mathcal{X}} = \times_{j=1}^{d} \mathbf{x}_j$, where $\mathbf{x}_j = (x_{i_j})_{i_j=1}^{n_j}$ for $j = 1, \ldots, d$. We denote $\boldsymbol{\mathcal{A}}(i_1, \ldots, i_d) = f(x_{i_1}, \ldots, x_{i_d})$ and abbreviate the d-dimensional tensor by $\boldsymbol{\mathcal{A}} = f(\boldsymbol{\mathcal{X}})$.

In the special case of $d = 2$, $\boldsymbol{\mathcal{A}}$ reduces to a matrix $\mathbf{A}$. A decomposition of this matrix can be obtained through the singular value decomposition (SVD):

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T. \tag{2.1}$$

Such a decomposition always exists and, since $\mathbf{A}$ is a real valued matrix, the SVD is unique up to sign change [42]. The SVD can be used to obtain a low-rank approximation of $\mathbf{A}$ by truncation of the singular values on the diagonal of $\boldsymbol{\Sigma}$. Unfortunately the SVD, as it is, can not be generalized to decompose tensors of dimension $d > 2$. Several approaches to this problem have been proposed over the years [26, 4, 18]. Amongst them the most popular are by far the canonical decomposition (CANDECOMP) [6, 22], the Tucker decomposition [43] and the tensor-train decomposition [31].

2.1. Classical tensor decompositions. The *canonical decomposition* aims to obtain an approximation of $\boldsymbol{\mathcal{A}}$ in terms of a sum of outer products:

$$\boldsymbol{\mathcal{A}} \simeq \boldsymbol{\mathcal{A}}_{CD} = \sum_{i=1}^{r} \mathbf{A}_i^{(1)} \circ \cdots \circ \mathbf{A}_i^{(d)} \,, \tag{2.2}$$

where $\mathbf{A}_i^{(k)}$ is the i-th column of matrix $\mathbf{A}^{(k)} \in \mathbb{R}^{n_k \times r}$ and $\circ$ denotes the outer product of two vectors. The upper bound of summation r is called the canonical rank of the tensor $\boldsymbol{\mathcal{A}}_{CD}$. The canonical decomposition is unique under mild conditions [38]. On the other hand a best rank-r decomposition – where we truncate the expansion similarly to the SVD case – does not always exist since the space of rank-r tensors is not closed [29, 39]. The computation of the canonical decomposition is based on the alternating least squares (ALS) method that, however, is not guaranteed to find a global minimum of the approximation error and has a number of other drawbacks and corresponding workarounds [26].

The *Tucker decomposition* is defined as follows:

$$\boldsymbol{\mathcal{A}} \simeq \sum_{i_1=1}^{r_1} \cdots \sum_{i_d=1}^{r_d} g_{i_1 \dots i_d} \left(\mathbf{A}_{i_1}^{(1)} \circ \cdots \circ \mathbf{A}_{i_d}^{(d)} \right), \tag{2.3}$$

where the *core tensor* $\boldsymbol{\mathcal{G}}$, defined by $\boldsymbol{\mathcal{G}}(i_1 \dots i_d) = g_{i_1 \dots i_d}$, accounts for weighting interactions between different components in different dimensions. This expansion is not unique, due to the possibility of applying rotations on the core tensor and their inverse on the components $\mathbf{A}^{(i)}$. However, the chances of obtaining an unique decomposition can be improved if sparsity is imposed on the core tensor (see [26] and the references therein). The Tucker decomposition is stable, but the number of parameters to be determined grows exponentially with the dimension d of the tensor due to the presence of the core tensor $\boldsymbol{\mathcal{G}}$. This limits the applicability of Tucker decomposition to "low" dimensional problems.

2.2. Tensor-train decomposition. The limited applicability of the Tucker decomposition to low-dimensional problems can be overcome using a hierarchical singular value decomposition, where the function is not decomposed with a single core $\boldsymbol{\mathcal{G}}$ that relates each dimension, but with a hierarchical tree of cores – usually binary – that relates a couple of dimensions at a time. This approach goes under the name of hierarchical Tucker or $\mathcal{H}$-Tucker decomposition [17]. A particular type of $\mathcal{H}$-Tucker decomposition is the tensor-train decomposition, which retains many of the characteristics of the $\mathcal{H}$-Tucker decomposition, but with a simplified formulation (see [17, Sec 5.3] for a comparison). The tensor-train decomposition has the following properties that make it attractive:

- existence of the full-rank approximation [31, Thm 2.1],
- existence of the low-rank best approximation [31, Cor 2.4],
- an algorithm that returns a sub-optimal TT-approximation (see (2.7) and [31, Cor 2.4]),
- memory complexity that scales linearly with the dimension d [31, Sec 3],
- straightforward multi-linear algebra operations,
- a sampling algorithm for the construction of the TT-approximation with a computational complexity that scales linearly with the dimensionality [35].

DEFINITION 2.1 (Discrete tensor-train approximation). *Let $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, with entries $\boldsymbol{\mathcal{A}}(i_1, \dots, i_d)$. The TT-rank $\mathbf{r} = (r_0, \dots, r_d)$ approximation of $\boldsymbol{\mathcal{A}}$ is $\boldsymbol{\mathcal{A}}_{TT} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ defined by:*

$$\begin{aligned} \boldsymbol{\mathcal{A}}(i_1, \dots, i_d) &= \boldsymbol{\mathcal{A}}_{TT}(i_1, \dots, i_d) + \boldsymbol{\mathcal{E}}_{TT}(i_1, \dots, i_d) \\ &= \sum_{\alpha_0, \dots, \alpha_d = 1}^{\mathbf{r}} G_1(\alpha_0, i_1, \alpha_1) \cdots G_d(\alpha_{d-1}, i_d, \alpha_d) + \boldsymbol{\mathcal{E}}_{TT}(i_1, \dots, i_d) \,, \end{aligned} \tag{2.4}$$

where $\boldsymbol{\mathcal{E}}_{TT}$ is the residual term and $r_0 = r_d = 1$.

We can see that each of the cores G_i is "connected" to the adjacent cores G_{i-1} and G_{i+1} by the indices α_{i-1} and α_i. This property led to the name tensor-train decomposition.

It can be proved [31] that the TT-approximation is exact ($\boldsymbol{\mathcal{E}}_{TT} = \mathbf{0}$) for

$$r_k = \text{rank}\left(\mathbf{A}_k\right) \quad , \qquad \forall\, k \in \{1, \dots, d\}\ , \tag{2.5}$$

where $\mathbf{A}_k$ is the k-th unfolding of $\boldsymbol{\mathcal{A}}$, that corresponds to the MATLAB/NumPy operation:

$$\mathbf{A}_k = \text{reshape}\left(\boldsymbol{\mathcal{A}}, \prod_{s=1}^{k} n_s, \prod_{s=k+1}^{d} n_s\right). \tag{2.6}$$

Furthermore if $r_k \leq \text{rank}\mathbf{A}_k$, the TT-rank $\mathbf{r}$ approximation $\boldsymbol{\mathcal{A}}^{\text{best}}$, which is optimal in the Frobenius norm, always exists and the algorithm `TT-SVD` [31] produces a quasi-optimal approximation to it. In particular, if $\boldsymbol{\mathcal{A}}_{TT}$ is the numerical approximation of $\boldsymbol{\mathcal{A}}$ obtained with `TT-SVD`, then

$$\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_{TT}\|_F \leq \sqrt{d-1}\|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}^{\text{best}}\|_F \quad . \tag{2.7}$$

Assuming that the TT-ranks are all equal $r_k = \text{rank}\mathbf{A}_k = r$, and that $n_1 = \ldots = n_d = n$, the TT-decomposition $\boldsymbol{\mathcal{A}}_{TT}$ requires the storage of $\mathcal{O}\left(dnr^2\right)$ parameters. Thus the representation (2.4) scales linearly with the dimension. A further reduction in the required storage can be achieved using the *Quantics* TT-format [33, 25], which, for $n = 2^m$, leads to $\mathcal{O}\left(dmr^2\right)$.

The computational complexity of the `TT-SVD` depends on the selected accuracy, but for $r_k = \text{rank}\mathbf{A}_k = r$ and $n_1 = \ldots = n_d = n$, the number of flops required by the algorithm are $\mathcal{O}\left(rn^d\right)$. We see that the computational complexity grows exponentially with the dimension and thus the *curse of dimensionality* is not resolved, except for the memory complexity of the final compressed representation. At this stage it is important to point out that the simpler formulation of tensor-train against the more complex $\mathcal{H}$-Tucker decomposition gives away the possibility of implementing a parallel version of `TT-SVD` [17] and gaining a factor $1/\log_2(d)$ in the computational complexity. However, this would not resolve the exponential growth of the computational complexity with respect to the dimensionality. In any case, `TT-SVD` is not suitable to be used for high-dimensional problems, because it first requires the storage of the full tensor. This means that the initial memory requirement scales exponentially with the dimension of the problem. In the next section we will discuss a method to construct an approximation to the tensor using a small number of function evaluations.

An open question in tensor-train decomposition regards the ordering of the indices of $\boldsymbol{\mathcal{A}}$. Different orderings of the d indices of $\boldsymbol{\mathcal{A}}$ can lead to higher or lower TT-ranks. As a consequence the memory efficiency changes depending on this ordering. We would like to have the pairs of indices for which $\boldsymbol{\mathcal{A}}$ is high-rank close to each other, so that the rank will be high only for the cores connecting these pairs. If this doesn't happen, the non-separability of a pair of dimensions will be carried on from core to core, making the decomposition more expensive. We will point to several examples where this problem arises in Section 5.3.

2.3. TT-DMRG-cross. The `TT-SVD` algorithm is expensive and, more importantly, requires the evaluation of the function on a full tensor grid. An alternative approach to the `TT-SVD` is provided by the `TT-DMRG-cross` algorithm (see [35] for a detailed description). This method hinges on the idea of the Density Matrix Renormalization Group [45] (DMRG) and of matrix skeleton decomposition [16]. For $d = 2$ and $\mathbf{A} \in \mathbb{R}^{m\times n}$, the skeleton decomposition is defined by:

$$\mathbf{A} \simeq \mathbf{A}(:, \mathcal{J})\mathbf{A}(\mathcal{I}, \mathcal{J})^{-1}\mathbf{A}(\mathcal{I}, :)\ , \tag{2.8}$$

where $\mathcal{I} = (i_1, \dots, i_r)$ and $\mathcal{J} = (j_1, \dots, j_r)$ are subset of the index sets $[1, \dots, m]$ and $[1, \dots, n]$. The selection of the indices $(\mathcal{I}, \mathcal{J})$ need to be such that most of the information contained in $\mathbf{A}$ is carried on through the decomposition. It turns out that the optimal submatrix $\mathbf{A}(\mathcal{I}, \mathcal{J})$ is the one

with maximal determinant in modulus among all the $r \times r$ submatrices of $\mathbf{A}$ [15]. The problem of finding such a matrix turns out to be NP-hard [7]. An approximation to the solution of this problem can be found using the `maxvol` algorithm [15], in an row-column alternating fashion as explained in [32]. Running `maxvol` is computationally inexpensive and requires $2c(n-r)r$ operations, where c is a usually small constant in many practical applications.

In practice the problem of finding the TT-decomposition $\boldsymbol{\mathcal{A}}_{TT}$, can be shaped as the minimization problem:

$$\min_{G_1,\dots,G_d} \|\boldsymbol{\mathcal{A}} - \boldsymbol{\mathcal{A}}_{TT}\|_F \quad . \tag{2.9}$$

One possible approach for solving this problem is `TT-cross` [32]. Here the optimization is performed through *left-to-right* and *right-to-left* sweeps of the cores and using the matrix skeleton decomposition in order to find the most relevant fibers in the d dimensional space. A *fiber* is, for a d-dimensional tensor $\boldsymbol{\mathcal{A}}$, the equivalent of what rows and columns are for a matrix. In MATLAB/NumPy notation, the $(i_1, \dots, i_{k-1}, i_{k+1}, \dots, i_d)$ fiber along the k-th dimension is $\boldsymbol{\mathcal{A}}(i_1, \dots, i_{k-1}, :, i_{k+1}, \dots, i_d)$. This approach provides linear scaling in the number of entries evaluated. On the other hand, it requires the TT-ranks to be known a priori in order to select the right number of fibers for each dimension. The underestimation of these ranks leads to a poor (and in some cases erroneous) approximation, while an overestimation of them leads to an increased computational effort.

A more viable approach is the `TT-DMRG-cross` [35], where the optimization is done over two cores G_k, G_{k+1} at a time. In practice at step k of the sweeps, the core $W_k(i_k, i_{k+1}) = G_k(i_k)G_k(i_{k+1})$ solving (2.9) is found, and the cores G_k and G_{k+1} are recovered through SVD. The identification of the relevant core W_k is again performed using the maximum volume principle, aiming at the selection of the most important planes $\boldsymbol{\mathcal{A}}(i_1, \dots, i_{k-1}, :, :, i_{k+2}, \dots, i_d)$ in the d-dimensional space. Differently from `TT-cross`, this method is *rank revealing*, meaning that the TT-ranks do not need to be guessed a priori. This means that the method is able to determine them automatically.

3. Relevant results from approximation theory. The main objective of this work is to extend the TT-format to be used for the construction of a surrogate function of f. To do this we need to consider the case where some smoothness can be assumed on f. Here we will review some concepts from polynomial approximation theory which, in subsequent sections, will be combined with the tensor-train decomposition. In the following, we will make use of the Sobolev spaces:

$$\mathcal{H}^k_\mu(\mathbf{I}) = \left\{ f \in L^2_\mu(\mathbf{I}) : \sum_{|\mathbf{i}|\le k} \|D^{(\mathbf{i})} f\|_{L^2_\mu(\mathbf{I})} < +\infty \right\} , \tag{3.1}$$

where $k \ge 0$, $D^{(\mathbf{i})} f$ is the $\mathbf{i}$-th weak derivative of f, $\mathbf{I} = I_1 \times \cdots \times I_d$ is the product of intervals of $\mathbb{R}$ and $\mu : \mathcal{B}(\mathbf{I}) \to \mathbb{R}$ is a σ-finite measure on the Borel σ-algebra defined on $\mathbf{I}$. This space is equipped with the norm $\|\cdot\|^2_{\mathcal{H}^k_\mu(\mathbf{I})}$ defined by

$$\|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} = \sum_{|\mathbf{i}|\le k} \|D^{(\mathbf{i})} f\|^2_{L^2_\mu(\mathbf{I})} \tag{3.2}$$

and the semi-norm $|\cdot|_{\mathbf{I},\mu,k}$ defined by

$$|f|^2_{\mathbf{I},\mu,k} = \sum_{|\mathbf{i}|=k} \|D^{(\mathbf{i})} f\|^2_{L^2_\mu(\mathbf{I})}. \tag{3.3}$$

3.1. Projection. A function $f \in L^2_\mu(\mathbf{I})$ can be approximated by the expansion based on its projection onto a set of basis for $L^2_\mu(\mathbf{I})$. The following results hold both for compact and non-compact supports $\mathbf{I}$.

DEFINITION 3.1 (Spectral expansion). *Let $\mathbf{I} \subseteq \mathbb{R}^d$ and $f \in L^2_\mu(\mathbf{I})$ be a square integrable function with respect to the measure μ. Let $\{\Phi_{\mathbf{i}}\}_{|\mathbf{i}|=0}^{\infty}$ be a set of orthonormal polynomials forming a basis*

for $L^2_\mu(\mathbf{I})$*, where* $\Phi_{\mathbf{i}}(\mathbf{x}) = \phi_{i_1,1}(x_1)\cdots\phi_{i_d,d}(x_d)$ *and* $\mathbf{i} = (i_1,\dots,i_d)$*. For* $\mathbf{N} = (N_1,\dots,N_d)$*, the truncated spectral expansion of degrees* $\mathbf{N}$ *of* f *is defined in terms of the projection operator* $P_{\mathbf{N}} : L^2_\mu(\mathbf{I}) \to \text{span}(\{\Phi_{\mathbf{i}}\}_{\mathbf{i}=0}^{\mathbf{N}})$ *defined by*

$$P_{\mathbf{N}} f = \sum_{0\le\mathbf{i}\le\mathbf{N}} c_{\mathbf{i}}\Phi_{\mathbf{i}}, \qquad c_{\mathbf{i}} = \int_{\mathbf{I}} f\Phi_{\mathbf{i}} d\mu(\mathbf{x}). \tag{3.4}$$

where $\mathbf{i} \le \mathbf{N}$ *denotes* $\bigwedge_{1\le j\le d}(\mathbf{i}_j \le \mathbf{N}_j)$.

For simplicity, in the following we define $P_N := P_{\mathbf{N}}$ when $N_1 = \ldots = N_d = N$. The rate of convergence of the spectral expansion (3.4) is determined by the smoothness of f.

Proposition 3.2 (Convergence of spectral expansion [23, 5]). *For* $k \ge 0$*, let* $f \in \mathcal{H}^k_\mu(\mathbf{I})$*, then*

$$\|f - P_N f\|_{L^2_\mu(\mathbf{I})} \le C(k) N^{-k} |f|_{\mathbf{I},\mu,k} \qquad . \tag{3.5}$$

In practice the coefficients $c_{\mathbf{i}}$, see (3.4), are approximated using discrete inner products based on quadrature rules of sufficient accuracy. We will focus here on the use of high-order accurate Gauss-type quadrature rules [10], defined by the points and weights $(z_{\mathbf{i}}, w_{\mathbf{i}})_{\mathbf{i}=0}^{\mathbf{N}}$, where $(z_{\mathbf{i}})_{\mathbf{i}=0}^{\mathbf{N}} \subset \mathbf{I} = I_1 \times \cdots \times I_d$. These points and weights can be readily obtained using the Golub-Welsch algorithm [14]. A d-dimensional integral can then be approximated by:

$$\int_{\mathbf{I}} f(\mathbf{x}) d\mu(\mathbf{x}) \approx \sum_{\mathbf{i}=0}^{\mathbf{N}} f(z_{\mathbf{i}}) w_{\mathbf{i}} =: U_{\mathbf{N}}(f) \quad . \tag{3.6}$$

Gauss, Gauss-Radau and Gauss-Lobatto quadrature rules can be applied to open intervals, intervals open on one side and closed intervals respectively, being exact for functions f of polynomial orders up to $2\mathbf{N}+1$, $2\mathbf{N}$ and $2\mathbf{N}-1$ respectively. The discrete version of the spectral expansion (3.4) is then given by the following definition.

Definition 3.3 (Discrete projection). *Let* $(\mathbf{z_i}, w_{\mathbf{i}})_{\mathbf{i}=0}^{N}$ *be a set of quadrature points and weights. The discrete projection of* f *is defined in terms of the operator* $\tilde{P}_{\mathbf{N}} : L^2_\mu(\mathbf{I}) \to \text{span}(\{\Phi_{\mathbf{i}}\}_{\mathbf{i}=0}^{\mathbf{N}})$*, defined by*

$$\tilde{P}_{\mathbf{N}} f = \sum_{\mathbf{i}=0}^{\mathbf{N}} \tilde{c}_{\mathbf{i}}\Phi_{\mathbf{i}}, \qquad \tilde{c}_{\mathbf{i}} = U_{\mathbf{N}}(f\Phi_{\mathbf{i}}) = \sum_{\mathbf{i}=0}^{\mathbf{N}} f(\mathbf{z_i})\Phi_{\mathbf{i}}(\mathbf{z_i}) w_{\mathbf{i}} \quad . \tag{3.7}$$

If the quadrature rule is a Gauss quadrature rule, then the discrete projection will be exact for $f \in \mathbb{P}_{\mathbf{N}}$, the set of polynomials of degree up to $\mathbf{N}$.

3.2. Interpolation. A function f can also be approximated using interpolation on a set of nodes and assuming a certain level of smoothness in between them. Here we will consider the piecewise linear interpolation and polynomial interpolation on closed and bounded domains $\mathbf{I} = I_1 \times \cdots \times I_d$. Other interpolation rules could be used inside the same framework for specific problems.

The linear interpolation of function $f : [a,b] \to \mathbb{R}$ is based on an expansion in terms of basis functions called hat functions: given a set of distinct ordered nodes $\{x_i\}_{i=0}^N \in [a,b]$ with $x_0 = a$ and $x_N = b$, the hat functions are:

$$e_i(x) = \begin{cases} \frac{x - x_{i-1}}{x_i - x_{i-1}} & \text{if } \mathrm{x_{i-1}} \le \mathrm{x} \le \mathrm{x_i} \wedge \mathrm{x} \ge \mathrm{a} \\ \frac{x - x_{i+1}}{x_i - x_{i+1}} & \text{if } \mathrm{x_i} \le \mathrm{x} \le \mathrm{x_{i+1}} \wedge \mathrm{x} \le \mathrm{b} \\ 0 & \text{otherwise} \end{cases} \quad . \tag{3.8}$$

When dealing with multiple dimensions several options are available. A very common choice of basis functions have their support over simplexes around a node. This allows the basis functions

$e_{\mathbf{i}}$ to be still linear. In this work we will instead choose basis functions that have their support on the hypercubes adjacent to a node. The basis functions $e_{\mathbf{i}}$ cannot be linear anymore in order to attain the linear interpolation property: they need to be bilinear in two dimensions, trilinear in three dimensions and so on. Letting V be the set of piecewise continuous functions on $\mathbf{I}$, the multi-linear interpolation $I_{\mathbf{N}} : V \to \mathcal{C}^0(\mathbf{I})$ is then defined by

$$I_{\mathbf{N}} f(\mathbf{x}) = \sum_{\mathbf{i}=0}^{\mathbf{N}} \hat{c}_{\mathbf{i}} e_{\mathbf{i}}(\mathbf{x}), \qquad \hat{c}_{\mathbf{i}} = f\left(\mathbf{x}_{\mathbf{i}}\right), \tag{3.9}$$

where $\{\mathbf{x_i}\}_{\mathbf{i}=0}^{\mathbf{N}} = \{x_i^1\}_{i=0}^{N_1} \times \cdots \times \{x_i^d\}_{i=0}^{N_d}$ is a tensor grid of points. Again we will use the notation $I_N := I_{\mathbf{N}}$ when $N_1 = \ldots = N_d = N$. Assuming that the grid points are uniformly distributed, the convergence of the approximation is given by:

PROPOSITION 3.4 (Convergence of linear interpolation [3]). *Let $f \in \mathcal{H}^2_\mu(\mathbf{I})$, then*

$$\|f - I_N f\|_{L^2_\mu(\mathbf{I})} \leq C N^{-2} |f|_{\mathbf{I},\mu,2} \quad . \tag{3.10}$$

The second type of interpolation that we want to discuss within this work is the Lagrange interpolation. This is based on the Lagrange polynomials $\{l_i\}_{i=1}^N$, defined by

$$l_i(x) = \prod_{\substack{0 \leq m < k \\ m \neq i}} \frac{x - x_m}{x_i - x_m}, \tag{3.11}$$

where $\{x_i\}_{i=1}^k \in [a, b]$ are non uniformly distributed nodes, such as the Gauss nodes introduced in section 3.1. This choice is made in order to avoid the *Runge phenomenon* and assure an accurate approximation. The polynomial interpolation $\Pi_N : V \to \text{span}\left(\{l_i\}_{i=0}^N\right)$ is given by

$$\Pi_N f(x) = \sum_{i=0}^{N} \hat{c}_i l_i(x), \qquad \hat{c}_i = f\left(x_i\right). \tag{3.12}$$

The Lagrange interpolation has many theoretical issues when applied to the interpolation of multivariate functions. However, in the scope of this paper, we will only consider tensor grids of nodes, for which such approximation has no theoretical issue. As we will see in the next sections, such tensor grids of nodes will be never constructed explicitly thanks to the usage of the tensor-train decomposition, but the convergence properties of the Lagrange interpolation on tensor grids will be useful for analysis purposes.

The convergence of the Lagrange interpolation is again dictated by the smoothness of the functional that is approximated.

PROPOSITION 3.5 (Convergence of Lagrange interpolation [2, 5]). *For $k \geq 1$, let $f \in \mathcal{H}^k_\mu(\mathbf{I})$, then*

$$\|f - \Pi_N f\|_{L^2_\mu(\mathbf{I})} \leq C(k) N^{-k} |f|_{\mathbf{I},\mu,k} \quad . \tag{3.13}$$

4. Spectral tensor-train decomposition. Now we blend the discrete tensor-train decomposition of Section 2.2 with the polynomial approximations described in Section 3. First, we construct a continuous version of the tensor-train decomposition, termed the *functional* tensor-train (FTT) decomposition. The construction proceeds by recursively decomposing non-symmetric square integrable kernels through auxiliary symmetric square integrable kernels, as in Schmidt [36]. Next, we prove that the decomposition converges under some regularity conditions and that the cores of the

FTT-decomposition inherit the regularity properties of the original function, and thus are amenable to spectral approximation when the original function is smooth. Based on this analysis, we propose an efficient approach to high-dimensional function approximation that employs one-dimensional polynomial approximations of the cores of the FTT-decomposition, and we analyze the convergence of these approximations.

4.1. Functional tensor-train decomposition. Let $X \times Y \subset \mathbb{R}^d$ and let f be a Hilbert-Schmidt kernel with respect to the finite measure $\mu : \mathcal{B}(X \times Y) \to \mathbb{R}$, i.e. $f \in L^2_\mu(X \times Y)$. We restrict our attention to product measures, so $\mu = \mu_x \times \mu_y$. The operator

$$\begin{aligned} T : L^2_{\mu_y}(Y) &\to L^2_{\mu_x}(X) \\ g &\mapsto \int_Y f(x,y)g(y)d\mu_y(y) \end{aligned} \tag{4.1}$$

is linear, bounded and compact [19, Cor. 4.6]. The Hilbert adjoint operator of T is $T^* : L^2_{\mu_x}(X) \to L^2_{\mu_y}(Y)$. Then $TT^* : L^2_{\mu_x}(X) \to L^2_{\mu_x}(X)$ is a compact Hermitian operator. By the Spectral Theory on compact operators the spectrum of TT^* is only formed by a countable set of eigenvalues and the only point of accumulation is zero [28, Thm 8.3-1,8.6-4]. Being self-adjoint, the eigenfunctions $\{\psi(x;(i))\}_{i=1}^\infty \subset L^2_{\mu_x}(X)$ corresponding to the eigenvalues of TT^* form an orthonormal basis [19, Cor. 4.7]. Also $T^*T : L^2_{\mu_y}(Y) \to L^2_{\mu_y}(Y)$ is a self-adjoint compact operator with eigenfunctions $\{\phi(y;(i))\}_{i=1}^\infty \subset L^2_{\mu_y}(Y)$. Then we have the following expansion of f.

DEFINITION 4.1 (Functional-SVD). *Given the set of eigenvalues $\{\lambda(i)\}_{i=1}^\infty$ and the set of eigenfunctions $\{\psi(x;(i))\}_{i=1}^\infty$ and $\{\phi(y;(i))\}_{i=1}^\infty$, of the integral operators TT^* and T^*T respectively, the functional-SVD of f is:*

$$f = \sum_{i=1}^\infty \sqrt{\lambda(i)}\psi(\,\cdot\,;(i)) \otimes \phi(\,\cdot\,;(i))\,. \tag{4.2}$$

In the general setting considered the convergence of (4.2) is in L^2_μ.

Let now $I_1 \times \cdots \times I_d = \mathbf{I} \subseteq \mathbb{R}^d$ and let f be a Hilbert-Schmidt kernel with respect to the finite measure $\mu : \mathcal{B}(\mathbf{I}) \to \mathbb{R}$, i.e. $f \in L^2_\mu(\mathbf{I})$. We assume $\mu = \prod_{i=1}^d \mu_i$. Applying the functional-SVD to f with $X = I_1$ and $Y = I_2 \times \ldots \times I_d$, we obtain

$$f(\mathbf{x}) = \sum_{i_1=1}^\infty \sqrt{\lambda(i_1)}\psi_1\left(x_1;(i_1)\right)\phi_1\left(x_2,\ldots,x_d;(i_1)\right)\,. \tag{4.3}$$

If the functional-SVD is now applied to $\{\phi_1(x_2,\ldots,x_d;(i_1))\}_{i_1=1}^\infty$ with $X = I_2$ and $Y = I_3 \times \ldots \times I_d$, we get

$$f(\mathbf{x}) = \sum_{i_1=1}^\infty \sqrt{\lambda(i_1)}\psi_1(x_1;(i_1)) \sum_{i_2=1}^\infty \sqrt{\lambda(i_1,i_2)}\psi_2(x_2;(i_1,i_2))\phi_2(x_3,\ldots,x_d;(i_1,i_2))\,. \tag{4.4}$$

Proceeding inductively to $d-1$, we obtain the separated representation of f:

$$f(\mathbf{x}) = \sum_{i_1,\ldots,i_{d-1}=1}^\infty \sigma(i_1,\ldots,i_{d-1})\psi_1\left(x_1;(i_1)\right)\cdots\psi_d\left(x_d;(i_1,\ldots,i_{d-1})\right) \quad . \tag{4.5}$$

where $\sigma(i_1,\ldots,i_{d-1}) = \sqrt{\lambda(i_1)}\cdots\sqrt{\lambda(i_1,\ldots,i_{d-1})}$ are the singular values of the decomposition. If we define $\gamma_j\left((i_0,\ldots,i_{j-1});x_j;(i_0,\ldots,i_j)\right) := \psi_j\left(x_j;(i_1,\ldots,i_j)\right)$ and let α_j be a one long index for (α_{j-1},i_j), we can rewrite the sums in a convenient compact format: for $\alpha_0 = \alpha_d = 1$ and $\boldsymbol{\alpha} = (\alpha_0,\ldots,\alpha_d)$, we can write:

$$f(\mathbf{x}) = \sum_{\alpha_1,\ldots,\alpha_{d-1}=1}^\infty \sigma(\boldsymbol{\alpha})\cdot\gamma_1(\alpha_0,x_1,\alpha_1)\cdots\gamma_d(\alpha_{d-1},x_d,\alpha_d)\,. \tag{4.6}$$

We will call this format, the *functional* tensor-train (FTT) decomposition. Note that in the overloading of the notation in (4.6) the ordering of the sums was not changed.

In contrast with the definition of the discrete TT-approximation (see Def. 2.1), here we chose to carry the singular values in the d-dimensional tensor $\sigma(\boldsymbol{\alpha})$ for analysis purposes. We could have removed them from the final formulation by multiplying the singular values by their singular functions during the construction of the decomposition.

If we now apply a truncation to such expansion we obtain the functional format of the tensor-train approximation.

DEFINITION 4.2 (FTT-approximation). *Let $I_1 \times \cdots \times I_d = \mathbf{I} \subseteq \mathbb{R}^d$ and $f \in L^2_\mu(\mathbf{I})$. For $\mathbf{r} = (1, r_1, \ldots, r_{d-1}, 1)$, a TT-rank $\mathbf{r}$ functional TT-approximation of f is:*

$$f_{TT}(\mathbf{x}) := \sum_{\alpha_0,\ldots,\alpha_d=1}^{\mathbf{r}} \sigma(\boldsymbol{\alpha}) \cdot \gamma_1(\alpha_0, x_1, \alpha_1) \cdots \gamma_d(\alpha_{d-1}, x_d, \alpha_d) \,, \tag{4.7}$$

where $\gamma_i(\alpha_{i-1}, \cdot, \alpha_i) \in L^2_{\mu_i}$ and $\langle \gamma_k(\alpha_{k-1}, \cdot, (\alpha_{k-1}, j)), \gamma_k(\alpha_{k-1}, \cdot, (\alpha_{k-1}, l)) \rangle_{L^2_{\mu_k}} = \delta_{jl}$. The residual of such approximation will be denoted by $R_{TT} := f - f_{TT}$.

We will call $\{\gamma_i\}_{i=1}^d$ the cores of the approximation in agreement with the notation used for the discrete tensor-train approximation.

PROPOSITION 4.3. *Let the functional tensor-train decomposition* (4.6) *be truncated retaining the biggest singular values $\sigma(\boldsymbol{\alpha})$, then the approximation* (4.7) *fulfills the condition:*

$$\|R_{TT}\|^2_{L^2_\mu} = \min_{\substack{g \in L^2_\mu \\ \mathrm{TT-ranks(g)}=\mathbf{r}}} \|f - g\|^2_{L^2_\mu} = \sum_{\alpha_1=r_1+1}^{\infty} \cdots \sum_{\alpha_{d-1}=r_{d-1}+1}^{\infty} \sigma^2(\boldsymbol{\alpha}) \,. \tag{4.8}$$

Proof. We first notice that exploiting the orthonormality of the cores, yields

$$\|R_{TT}\|^2_{L^2_\mu} = \sum_{\alpha_1=r_1+1}^{\infty} \cdots \sum_{\alpha_{d-1}=r_{d-1}+1}^{\infty} \sigma^2(\boldsymbol{\alpha}) \,. \tag{4.9}$$

The minimality is due to the construction of f_{TT} by a sequence of orthogonal projections that minimizes the error in the L^2_μ-norm. These projections are onto the subspaces spanned by the eigenfunctions of the Hermitian operators induced by the tensor f, and are thus optimal [40, 44]. ∎

The result given in proposition 4.3 does not directly involve any property of the function f. We try then to link this estimate with the regularity of f. To do so, we will use the following auxiliary result, which is a particular case of [37, Prop. 2.21] and the next two lemmas which are proved in appendix B.

PROPOSITION 4.4. *Let $\mathbf{I} \subset \mathbb{R}^d$ be a bounded domain and $V \in L^2_{\mu\otimes\mu}(\mathbf{I}\times\mathbf{I})$ be a symmetric kernel of the compact non-negative integral operator $\mathcal{V} : L^2_\mu(\mathbf{I}) \to L^2_\mu(\mathbf{I})$. If V is $\mathcal{H}^k_\mu(\mathbf{I}\times\mathbf{I})$ with $k > 0$ and $\{\lambda_m\}_{m\geq 1}$ denotes the eigenvalue sequence of $\mathcal{V}$, then*

$$\lambda_m \leq |V|_{\mathbf{I}\times\mathbf{I},\mu,k} m^{-k/d} \qquad \forall m \geq 1 \,. \tag{4.10}$$

LEMMA 4.5. *Let $f \in \mathcal{H}^k_\mu(\mathbf{I})$, $\bar{\mathbf{I}} = I_2 \times \ldots \times I_d$ and $J(x, \bar{x}) = \langle f(x, y), f(\bar{x}, y) \rangle_{L^2_\mu(\bar{\mathbf{I}})}$. Then, $J \in \mathcal{H}^k_\mu(I_1 \times I_1)$*

$$|J|_{I_1\times I_1,\mu,k} \leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \,. \tag{4.11}$$

LEMMA 4.6. *Let $f \in \mathcal{H}^k_\mu(\mathbf{I})$, $\bar{\mathbf{I}} = I_2 \times \ldots \times I_d$ and*

$$f_{TT}(x_1, \ldots, x_d) = \sum_{i_1=1}^{r_1} \sqrt{\lambda(i_1)} \psi_1(x_1; i_1) \phi_1(x_2, \ldots, x_d; i_1) \,. \tag{4.12}$$

Then,

$$\|\phi_1(i_1)\|^2_{\mathcal{H}^k_\mu(\bar{\mathbf{I}})} \leq \frac{1}{\lambda(i_1)}\|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \,. \tag{4.13}$$

In the sake of simplicity in the following analysis, we will let the ranks be $\mathbf{r} = (r, \dots, r)$.

THEOREM 4.7 (FTT-approximation convergence). *Let $f \in \mathcal{H}^k_\mu(\mathbf{I})$, then*

$$\|R_{TT}\|^2_{L^2_\mu} \leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})}\zeta(k, r+1)\frac{r^d - r}{r(r-1)} \qquad \textit{for } r > 1 \,, \tag{4.14}$$

where ζ is the Hurwitz Zeta function. Furthermore

$$\lim_{r\to\infty} \|R_{TT}\|^2_{L^2_\mu} \leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})}\frac{1}{(k-1)} \qquad \textit{for } k = d-1 \tag{4.15}$$

and

$$\lim_{r\to\infty} \|R_{TT}\|^2_{L^2_\mu} = 0 \qquad \textit{for } k > d-1 \,. \tag{4.16}$$

Proof. We start considering the case $\mathbf{I} = I_1 \times I_2 \times I_3$ and we define the following approximations of f, using the functional-SVD (4.2):

$$f_{TT,1} = \sum_{i_1=1}^{r_1} \sqrt{\lambda(i_1)}\psi_1(x_1; i_1)\phi_1(x_2, x_3; i_1) \,, \tag{4.17}$$

$$f_{TT} = \sum_{i_1=1}^{r_1} \sqrt{\lambda(i_1)}\psi_1(x_1; i_1)\phi_{TT,1}(x_2, x_3; i_1) \,, \tag{4.18}$$

where

$$\phi_{TT,1}(x_2, x_3; i_1) = \sum_{i_2=1}^{r_2} \sqrt{\lambda(i_1, i_2)}\psi_2(x_2; i_1, i_2)\phi_2(x_3; i_1, i_2) \,. \tag{4.19}$$

It is possible to show that $\langle f - f_{TT,1}, f_{TT,1} - f_{TT}\rangle_{L^2_\mu(\mathbf{I})} = 0$ and so

$$\|R_{TT}\|^2_{L^2_\mu(\mathbf{I})} = \|f - f_{TT}\|^2_{L^2_\mu(\mathbf{I})} = \|f - f_{TT,1}\|^2_{L^2_\mu(\mathbf{I})} + \|f_{TT,1} - f_{TT}\|^2_{L^2_\mu(\mathbf{I})} \,. \tag{4.20}$$

Exploiting the orthogonality of the singular functions, Proposition 4.4 and Lemma 4.5 we have

$$\|f - f_{TT,1}\|^2_{L^2_\mu(\mathbf{I})} = \sum_{i_1=r_1+1}^{\infty} \lambda(i_1) \leq \sum_{i_1=r_1+1}^{\infty} i_1^{-k}|J_0|_k \leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})}\zeta(k, r_1+1) \,, \tag{4.21}$$

where $J_0(x_1, \bar{x}_1) = \langle f(x_1, x_2, x_3), f(\bar{x}_1, x_2, x_3)\rangle_{L^2_\mu(I_2\times I_3)}$. Similarly:

$$\|\phi_1(i_1) - \phi_{TT,1}(i_1)\|^2_{L^2_\mu(I_2\times I_3)} \leq \sum_{i_2=r_2+1}^{\infty} i_2^{-k}|J_1(i_1)|_k \leq \|\phi_1(i_1)\|^2_{\mathcal{H}^k_\mu(\mathbf{I})}\zeta(k, r_2+1) \,, \tag{4.22}$$

where $J_1(x_2, \bar{x}_2; i_1) = \langle \phi_1(x_2, x_3; i_1), \phi_1(\bar{x}_2, x_3; i_1)\rangle_{L^2_\mu(I_3)}$. With the help of Lemma 4.6, this leads to

$$\begin{aligned}
\|f_{TT,1} - f_{TT}\|^2_{L^2_\mu(\mathbf{I})} &= \int_{\mathbf{I}} \left[\sum_{i_1=1}^{r_1} \sqrt{\lambda(i_1)}\psi_1(x_1; i_1)\left(\phi_1(x_2, x_3; i_1) - \phi_{TT,1}(x_2, x_3; i_1)\right)\right]^2 d\mu(\mathbf{x}) \\
&= \sum_{i_1=1}^{r_1} \lambda(i_1)\|\psi_1(i_1)\|^2_{L^2_\mu(I_1)}\|\phi_1(i_1) - \phi_{TT,1}(i_1)\|^2_{L^2_\mu(I_2\times I_3)} \\
&\leq \sum_{i_1=1}^{r_1} \lambda(i_1)\|\phi_1(i_1)\|^2_{\mathcal{H}^k_\mu(\mathbf{I})}\zeta(k, r_2+1) \leq r_1\|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})}\zeta(k, r_2+1) \,.
\end{aligned} \tag{4.23}$$

Thus we obtain the bound

$$\|R_{TT}\|^2_{L^2_\mu(\mathbf{I})} \leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \left[\zeta(k, r_1+1) + r_1\zeta(k, r_2+1)\right] . \tag{4.24}$$

Let now $\mathbf{I} = I_1 \times \ldots \times I_d$ and $\mathbf{r} = (r, \ldots, r)$, for $r \geq 2$, then

$$\begin{aligned} \|R_{TT}\|^2_{L^2_\mu(\mathbf{I})} &\leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \sum_{i=1}^{d-1} \left(\prod_{j=1}^{i-1} r_j\right) \zeta(k, r_i+1) \\ &= \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \zeta(k, r+1) \sum_{i=1}^{d-1} r^{i-1} = \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \zeta(k, r+1) \frac{r^d - r}{r(r-1)} . \end{aligned} \tag{4.25}$$

This proves the first part of the theorem.

Let us now study the asymptotic behavior of $\|R_{TT}\|^2_{L^2_\mu}$ as $r \to \infty$. For $k > 1$, we can use the bound:

$$\zeta(k, r+1) = \sum_{i=r+1}^{\infty} i^{-k} \leq \int_{r+1}^{\infty} i^{-k} di = \frac{(r+1)^{-(k-1)}}{(k-1)} . \tag{4.26}$$

Plugging this into (4.25) and considering its asymptotic behavior as $r \to \infty$, we obtain:

$$\begin{aligned} \|R_{TT}\|^2_{L^2_\mu(\mathbf{I})} &\leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \frac{(r+1)^{-(k-1)}}{(k-1)} \frac{r^d - r}{r(r-1)} \\ &\approx \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \frac{1}{(k-1)r^{k-1}} \frac{r^{d-1}}{r} = \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \frac{r^{d-1-k}}{k-1} . \end{aligned} \tag{4.27}$$

This leads to the two asymptotic estimates (4.15) and (4.16), completing the proof. ∎

4.2. Regularity of the FTT-decomposition. In order to apply the traditional polynomial approximation theory to the functional tensor-train decomposition, we need that such decomposition retains the same regularity of the original function. In particular, in the scope of the polynomial approximation theory presented in Section 3, we need the boundedness of the weak derivatives used for the definition of Sobolev spaces (3.1). With this perspective, we will need the absolute convergence almost everywhere of the FTT-decomposition. Smithies [40, Thm. 14] proved that a kind of integrated Hölder continuity with exponent $\alpha > 1/2$ is a sufficient condition for the absolute convergence almost everywhere (a.e.) of the functional-SVD. The condition required by Smithies is a generalization of the Hölder continuity a.e. [41], as we show in Appendix A. The Smithies' result can be extended by construction to the FTT-decomposition:

COROLLARY 4.8 (Absolute convergence almost everywhere). *Let $I_1 \times \cdots \times I_d = \mathbf{I} \subset \mathbb{R}^d$ be closed and bounded, and $f \in L^2_\mu(\mathbf{I})$ be a Hölder continuous function with exponent $\alpha > 1/2$. Then the FTT-decomposition* (4.6) *converges absolutely almost everywhere.*

Now we can prove that if f belongs to a certain Sobolev space, then also the cores of the FTT-decomposition will belong to the same Sobolev space.

THEOREM 4.9 (FTT-decomposition and Sobolev spaces). *Let $I_1 \times \cdots \times I_d = \mathbf{I} \subset \mathbb{R}^d$ be closed and bounded, and $f \in L^2_\mu(\mathbf{I})$ be a Hölder continuous function with exponent $\alpha > 1/2$ such that $f \in \mathcal{H}^k_\mu(\mathbf{I})$. Then the FTT-decomposition* (4.6) *is such that $\gamma_j(\alpha_{j-1}, \cdot, \alpha_j) \in \mathcal{H}^k_{\mu_j}(I_j)$ for all j, α_{j-1} and α_j.*

Proof. We will first show this property for the functional-SVD (4.2) of the Hölder ($\alpha > 1/2$) continuous function $f \in \mathcal{H}^k_\mu(X \times Y)$. First we want to show that

$$D^{\mathbf{i}} f = \sum_{j=1}^{\infty} \sqrt{\lambda_j} (D^{i_1}\psi_j \otimes D^{i_2}\phi_j) , \tag{4.28}$$

where $\mathbf{i} = (i_1, i_2)$. Since f is Hölder ($\alpha > 1/2$) continuous, (4.2) converges absolutely a.e. by Smithies [40], then we can define

$$\infty > g := \sum_{j=1}^{\infty} \left| \sqrt{\lambda_j}(\psi_j \otimes \phi_j) \right| \geq \left| \sum_{j=1}^{\infty} \sqrt{\lambda_j}(\psi_j \otimes \phi_j) \right| , \tag{4.29}$$

where the domination holds almost everywhere. The series (4.2) is convergent almost everywhere by Smithies [40]. By the definition of weak derivative, for all $\chi \in \mathcal{C}_c^\infty(X \times Y)$:

$$(-1)^{|\mathbf{i}|} \int_{X\times Y} D^{\mathbf{i}} f \chi d\mu = \int_{X\times Y} f \chi^{(\mathbf{i})} d\mu \, . \tag{4.30}$$

Thus this holds also for any $\chi = \chi_x \otimes \chi_y \in \mathcal{C}_c^\infty(X) \otimes \mathcal{C}_c^\infty(X)$. Using the dominated convergence theorem, we obtain:

$$\begin{aligned}
(-1)^{|\mathbf{i}|} \int_{X\times Y} D^{\mathbf{i}} f \chi d\mu &= \int_{X\times Y} f \chi^{(\mathbf{i})} d\mu = \int_{X\times Y} \left(\sum_{j=1}^{\infty} \sqrt{\lambda_j}(\psi_j \otimes \phi_j) \right) \chi^{(\mathbf{i})} d\mu \\
&= \sum_{j=1}^{\infty} \sqrt{\lambda_j} \int_{X\times Y} (\psi_j \otimes \phi_j) \chi^{(\mathbf{i})} d\mu = \sum_{j=1}^{\infty} \sqrt{\lambda_j} \int_{X\times Y} \left(\psi_j \chi_x^{(i_1)} \right) \otimes \left(\phi_j \chi_y^{(i_2)} \right) d\mu \\
&= \sum_{j=1}^{\infty} \sqrt{\lambda_j} \left((-1)^{i_1} \int_X D^{i_1} \psi_j \chi_x d\mu_x \right) \left((-1)^{i_2} \int_Y D^{i_2} \phi_j \chi_y d\mu_y \right) \, .
\end{aligned}$$

Thus (4.28) holds. Next we want to show that $f \in \mathcal{H}_\mu^k(X \times Y)$ implies $\|D^{i_1}\psi_j\|_{L_\mu^2(X)} < \infty$ and $\|D^{i_2}\phi_j\|_{L_\mu^2(Y)} < \infty$ for $i_1, i_2 \leq k$. Thanks to (4.28) and due to the orthonormality of $\{\phi_j\}_{j=1}^\infty$ we have that

$$D^{i_1}\psi_j = \frac{1}{\sqrt{\lambda_j}} \left\langle D^{(i_1,0)} f, \phi_j \right\rangle_{L_\mu^2(Y)} \, . \tag{4.31}$$

Using the Cauchy-Schwarz inequality:

$$\begin{aligned}
\left\| D^{i_1}\psi_j \right\|_{L_\mu^2(X)}^2 &= \left\| \frac{1}{\sqrt{\lambda_j}} \left\langle D^{(i_1,0)} f, \phi_j \right\rangle_{L_\mu^2(Y)} \right\|_{L_\mu^2(X)}^2 \\
&\leq \left| \frac{1}{\lambda_j} \right| \|\phi_j\|_{L_\mu^2(Y)}^2 \left\| D^{(i_1,0)} f \right\|_{L_\mu^2(X\times Y)}^2 < \infty \, ,
\end{aligned} \tag{4.32}$$

where the last bound is due to the fact that $\{\phi_j\}_{j=1}^\infty \subset L_\mu^2(Y)$ – see Eqs. (4.1) and (4.3) – and $D^{(i_1,0)} f \in L_\mu^2(X \times Y)$ because $i_1 \leq k$ and $f \in \mathcal{H}_\mu^k(X \times Y)$. In the same way $\left\| D^{i_2}\phi_j \right\|_{L_\mu^2(Y)} < \infty$ for all $i_2 \leq k$. It follows that $\{\psi_j\}_{j=1}^\infty \subset \mathcal{H}_\mu^k(X)$ and $\{\phi_j\}_{j=1}^\infty \subset \mathcal{H}_\mu^k(Y)$. The extension to the FTT-decomposition (4.6) follows by its construction in terms of repeated functional-SVDs. ∎

Remark 1. The results above have the limitation of holding for functions defined on closed and bounded domains. In many practical cases, however, functions are defined on the real line, equipped with a finite measure. To the author's knowledge, the corresponding result for such cases has not been proved in literature. The result by Smithies [40, Thm. 14] hinges on a result by Hardy and Littlewood [21, Thm. 10] on the convergence of Fourier series, and this is the only passage in the proof where the closedness and boundedness of the domain is explicitly used. A similar result for an orthogonal system in $L_\mu^2(-\infty, \infty)$, where μ is a finite measure, would be sufficient to extend

Smithies' result to the real line. For one of the numerical examples presented in the following (Sec. 5.5), we will assume that this result holds.

Other regularity properties can be proved, given different kinds of continuity of the function f. These properties are not strictly necessary in the scope of polynomial approximation theory, so we will state them without proof. The first regards the continuity of the cores of the FTT-decomposition and follows directly from Mercer's theorem [24].

PROPOSITION 4.10 (Continuity). *Let $I_1 \times \cdots \times I_d = \mathbf{I} \subset \mathbb{R}^d$, and $f \in L^2_\mu(\mathbf{I})$ be a continuous function with FTT-decomposition* (4.6)*. Then $\gamma_i(\alpha_{i-1}, \cdot, \alpha_i)$ are continuous for every i and α_i.*

The second property regards the strong derivatives of the cores of the FTT-decomposition. It requires the Lipschitz continuity of the function and then follows from a result on the uniform convergence of the functional-SVD by Hammerstein [20, 41].

THEOREM 4.11 (Differentiability). *Let $I_1 \times \cdots \times I_d = \mathbf{I} \subset \mathbb{R}^d$ be closed and bounded, and let $f \in L^2_\mu(\mathbf{I})$ be a Lipschitz continuous function such that $\frac{\partial^\beta f}{\partial x_1^{\beta_1} \cdots \partial x_d^{\beta_d}}$ exists and is continuous on $\mathbf{I}$ for $\beta = \sum_{i=1}^d \beta_i$. Then the FTT-decomposition* (4.6) *is such that $\gamma_k(\alpha_{k-1}, \cdot, \alpha_k) \in \mathcal{C}^{\beta_k}(I_k)$ for all k, α_{k-1} and α_k.*

4.3. Polynomial approximation of the FTT-decomposition. All the theory is now in place in order to blend the FTT-decomposition with the polynomial approximations described in Section 3. We will consider the projection and the interpolation approach separately.

4.3.1. Functional tensor-train projection. We can now deal with the spectral approximation of the functional version of the tensor-train decomposition. Let $f \in \mathcal{H}^k_\mu(\mathbf{I})$ and f_{TT} be its FTT-approximation. If the projector (3.4) is applied to the FTT-approximation of f we obtain:

$$
\begin{aligned}
& P_{\mathbf{N}} f_{TT} = \sum_{\mathbf{i}=0}^{\mathbf{N}} \tilde{c}_{\mathbf{i}} \Phi_{\mathbf{i}} \quad , \\
& \tilde{c}_{\mathbf{i}} = \int_{\mathbf{I}} f_{TT} \Phi_{\mathbf{i}} d\mu(\mathbf{x}) = \sum_{\alpha_0, \ldots, \alpha_d = 1}^{\mathbf{r}} \sigma(\boldsymbol{\alpha}) \beta_1(\alpha_0, i_1, \alpha_1) \cdots \beta_d(\alpha_{d-1}, i_d, \alpha_d) \quad , \\
& \beta_n(\alpha_{n-1}, i_n, \alpha_n) = \int_{I_n} \gamma_n(\alpha_{n-1}, x_n, \alpha_n) \phi_{i_n}(x_n) d\mu_n(x_n) \quad .
\end{aligned}
\tag{4.33}
$$

Thus, the spectral expansion of the function can be obtained as a projection of its cores onto one dimensional basis functions. Furthermore the tensor-train representation of the expansion coefficients $\boldsymbol{\mathcal{C}} := [c_{\mathbf{i}}]_{\mathbf{i}=0}^{\mathbf{N}}$ can be obtained as a projection of the cores onto the one-dimensional basis functions.

The projector $P_{\mathbf{N}}$ is replaced by the discrete projector $\tilde{P}_{\mathbf{N}}$ (see (3.7)) and the the cores $\{\beta_i\}_{i=1}^d$ can be approximated by

$$
\beta_n(\alpha_{n-1}, i_n, \alpha_n) \approx \hat{\beta}_n(\alpha_{n-1}, i_n, \alpha_n) = \sum_{j=0}^{N_n} \gamma_n(\alpha_{n-1}, x_n^{(j)}, \alpha_n) \phi_{i_n}(x_n^{(j)}) w_n^{(j)} \,,
\tag{4.34}
$$

where $\{x_n^{(j)}, w_n^{(j)}\}_{j=0}^{N_n}$ are properly selected quadrature nodes and weights, e.g. Gauss-type, for the n-th dimension. In practice $\gamma_n(\alpha_{n-1}, x_n^{(j)}, \alpha_n)$ are the cores of the FTT-approximation of f evaluated at properly selected quadrature points. Thus, they can be obtained directly from the discrete TT-approximation of f evaluated on the grid formed by those points. This approximation is obtained by the `TT-DMRG-cross` algorithm, leading to substantial computational savings. The algorithm for computing the tensor-train decomposition $\boldsymbol{\mathcal{C}}_{TT}$ of $\boldsymbol{\mathcal{C}}$ is detailed in Algorithm 1.

Once the `FTT-projection-construction` step is done, the approximation can be evaluated on an arbitrary point $\mathbf{y} = \{y_1, \ldots, y_d\} \in \mathbf{I}$. This is described in Algorithm 2.

Algorithm 1 `FTT-projection-construction`

Require: The function $f : \mathbf{I} \to \mathbb{R}$, the measure $\mu = \prod_{n=1}^{d} \mu_n$, the integers $\mathbf{N} = \{N_n\}_{n=1}^{d}$ denoting the polynomial orders of approximation.
Ensure: $\boldsymbol{\mathcal{C}}_{TT}(i_1, \dots, i_d) = \sum_{\alpha_0,\dots,\alpha_d=1}^{\mathbf{r}} \hat{\beta}_1(\alpha_0, i_1, \alpha_1) \cdots \hat{\beta}_d(\alpha_{d-1}, i_d, \alpha_d)$, the TT-decomposition of the tensor of expansion coefficients.
Construct the set of basis functions $\left\{ \{\phi_{i_n,n}\}_{i_n=0}^{N_n} \right\}_{n=1}^{d}$ with respect to μ
Determine the Gauss-type points and weights $\{(\mathbf{x}_n, \mathbf{w}_n)\}_{n=1}^{d}$, $\mathbf{x}_n = \{x_n^{(i)}\}_{i=0}^{N_n}$, $\mathbf{w}_n = \{w_n^{(i)}\}_{i=0}^{N_n}$
Construct the discrete TT-decomposition $\boldsymbol{\mathcal{A}}_{TT}$ of $f\left(\times_{j=1}^{d} \mathbf{x}_j\right)$ through `TT-DMRG-cross`
for $n := 1$ **to** d **do**
  for all $(\alpha_{n-1}, \alpha_n) \in [0, r_{n-1}] \times [0, r_n]$ **do**
    $\hat{\beta}_n(\alpha_{n-1}, i_n, \alpha_n) = \sum_{j=0}^{N_n} G_n(\alpha_{n-1}, j, \alpha_n) \phi_{i_n,n}(x_n^{(j)}) w_n^{(j)}$
  end for
end for
return $\left\{ \hat{\beta}_n(\alpha_{n-1}, i_n, \alpha_n) \right\}_{n=1}^{d}$

Algorithm 2 `FTT-projection-evaluation`

Require: The set of cores $\left\{ \hat{\beta}_n(\alpha_{n-1}, i_n, \alpha_n) \right\}_{n=1}^{d}$ obtained through `FTT-projection-construction` and a set of N_y points $\mathbf{Y} = \{\mathbf{y}_1, \dots, \mathbf{y}_d\} \subset \mathbf{I}$.
Ensure: The polynomial approximation $P_{\mathbf{N}} f_{TT}(\mathbf{y})$ of $f(\mathbf{y})$
for $n := 1$ **to** d **do**
  for all $(\alpha_{n-1}, \alpha_n) \in [0, r_{n-1}] \times [0, r_n]$ **do**
    $\hat{G}_n(\alpha_{n-1}, \cdot, \alpha_n) = \sum_{j=0}^{N_n} \hat{\beta}_n(\alpha_{n-1}, j, \alpha_n) \phi_{j,n}(\mathbf{y_n})$
  end for
end for
$\boldsymbol{\mathcal{B}}_{TT}(i_1, \dots, i_d) = \sum_{\alpha_0,\dots,\alpha_d=1}^{\mathbf{r}} \hat{G}_1(\alpha_0, i_1, \alpha_1) \cdots \hat{G}_d(\alpha_{d-1}, i_d, \alpha_d)$
return $P_{\mathbf{N}} f_{TT}(\mathbf{Y}) := \{\boldsymbol{\mathcal{B}}_{TT}(i, \dots, i)\}_{i=1}^{N_y}$

By theorems 4.7 and 4.9, the convergence of the spectral expansion depends on the regularity of f. For $k > d - 1$, let $f \in \mathcal{H}_{\mu}^{k}(\mathbf{I})$, then:

$$
\begin{aligned}
\|f - P_N f_{TT}\|_{L^2_\mu(\mathbf{I})} &\leq \|f - f_{TT}\|_{L^2_\mu(\mathbf{I})} + \|f_{TT} - P_N f_{TT}\|_{L^2_\mu(\mathbf{I})} \\
&\leq \|f\|_{\mathcal{H}^k_\mu(\mathbf{I})} \sqrt{\frac{(r+1)^{-(k-1)}}{k-1} \frac{r^{d-1} - 1}{r - 1}} + C(k) N^{-k} |f_{TT}|_{\mathbf{I},\mu,k} \, .
\end{aligned} \tag{4.35}
$$

This result shows that the convergence is driven by the selection of the rank r and the polynomial order N, and that it improves for functions with increasing regularity. Thus we can save computational time in the estimation of the expansion coefficients $\boldsymbol{\mathcal{C}}$ by (4.34) and obtain an approximation $P_{\mathbf{N}} f_{TT}$ which converges spectrally.

4.3.2. Functional Tensor-train interpolation. Function interpolation can be extended to tensors easily, and the tensor-train format can be exploited in order to save computational time. We will first consider the linear interpolation, see Section 3.2. Let $\boldsymbol{\mathcal{X}} = \times_{j=1}^{d} \mathbf{x}_j$ be a $N_x^{(1)} \times \cdots \times N_x^{(d)}$ grid of distinct candidate nodes where the function f can be evaluated and let $\boldsymbol{\mathcal{Y}} = \times_{j=1}^{d} \mathbf{y}_j$ be a $\prod_{j=1}^{d} N_y$ grid of points constructed using the coordinates of a set of N_y points $\mathbf{Y} = \{\mathbf{y}_1^{(j)}, \dots, \mathbf{y}_d^{(j)}\}_{j=1}^{N_y}$. The interpolating values $f(\mathbf{Y})$ can be computed using the interpolation operator (3.9) from the grid $\boldsymbol{\mathcal{X}}$

Algorithm 3 `FTT-interpolation-evaluation`

Require: The discrete TT-decomposition $\boldsymbol{\mathcal{A}}_{TT}$ of $f(\boldsymbol{\mathcal{X}})$ – possibly obtained by `TT-DMRG-cross` – where $\boldsymbol{\mathcal{X}} = \times_{j=1}^{d} \mathbf{x}_j$ is a grid of points constructed from $\left\{\{x_n^{(i)}\}_{n=1}^{N_x^{(i)}}\right\}_{i=1}^{d}$, and a set of N_y points $\mathbf{Y} = \{\mathbf{y}_1, \dots, \mathbf{y}_d\} \subset \mathbf{I}$

Ensure: The interpolated approximation $I_{\mathbf{N}} f_{TT}(\mathbf{Y})$ or $\Pi_{\mathbf{N}} f_{TT}(\mathbf{Y})$ of $f(\mathbf{Y})$

Construct list $\left\{L^{(i)}\right\}_{i=1}^{d}$ of $N_y \times N_x^{(i)}$ (linear or Lagrange) interpolation matrices from $\mathbf{x}_i$ to $\mathbf{y}_i$
for $n := 1$ **to** d **do**
  for all $(\alpha_{n-1}, \alpha_n) \in [0, r_{n-1}] \times [0, r_n]$ **do**
    $\hat{G}_n(\alpha_{n-1}, \cdot, \alpha_n) = L^{(n)} G_n(\alpha_{n-1}, \cdot, \alpha_n)$
  end for
end for
$\boldsymbol{\mathcal{B}}_{TT}(i_1, \dots, i_d) = \sum_{\alpha_0, \dots, \alpha_d = 1}^{\mathbf{r}} \hat{G}_1(\alpha_0, i_1, \alpha_1) \cdots \hat{G}_d(\alpha_{d-1}, i_d, \alpha_d)$
return $I_{\mathbf{N}} f_{TT}(\mathbf{Y}) := \{\boldsymbol{\mathcal{B}}_{TT}(i, \dots, i)\}_{i=1}^{N_y}$

to the grid $\boldsymbol{\mathcal{Y}}$

$$f(\boldsymbol{\mathcal{Y}}) \simeq (I_{\mathbf{N}} f)(\boldsymbol{\mathcal{Y}}) = \mathbf{E} f(\boldsymbol{\mathcal{X}}), \qquad \mathbf{E} = E^{(1)} \otimes \cdots \otimes E^{(d)}, \tag{4.36}$$

where $E^{(k)}$ is a $N_y \times N_x^{(k)}$ matrix defined by $E^{(k)}(i,j) = e_j^{(k)}(\mathbf{y}_k^{(i)})$ as in (3.8), and then extracting the values in its diagonal $f(\mathbf{Y}) \simeq \{(I_{\mathbf{N}} f)(\boldsymbol{\mathcal{Y}})_{i,\dots,i}\}_{i=1}^{N_y}$. This leads to the multi-linear interpolation on hyper-cubic elements. If we instead use the FTT-approximation f_{TT} in (4.36), we obtain

$$\begin{aligned} (I_{\mathbf{N}} f_{TT})(\boldsymbol{\mathcal{Y}}) &= \mathbf{E} f_{TT}(\boldsymbol{\mathcal{X}}) = \mathbf{E} \sum_{\alpha=0,\dots,\alpha_d=1}^{\mathbf{r}} \sigma(\boldsymbol{\alpha}) \gamma_1(\alpha_0, \mathbf{x}_1, \alpha_1) \cdots \gamma_d(\alpha_{d-1}, \mathbf{x}_d, \alpha_d) \\ &= \sum_{\alpha=0,\dots,\alpha_d=1}^{\mathbf{r}} \sigma(\boldsymbol{\alpha}) \beta_1(\alpha_0, \mathbf{y}_1, \alpha_1) \cdots \beta_d(\alpha_{d-1}, \mathbf{y}_d, \alpha_d) \ , \\ \beta_n(\alpha_{n-1}, \mathbf{y}_n, \alpha_n) &= E^{(n)} \gamma_n(\alpha_{n-1}, \mathbf{x}_n, \alpha_n) \ , \end{aligned} \tag{4.37}$$

where instead of working with the tensor-matrix $\mathbf{E}$, we can work with the more manageable matrices $\{E^{(i)}\}_{i=1}^{d}$ (see Alg. 3). The construction of the approximation in this case corresponds exactly to the application of the `TT-DMRG-cross` algorithm to $f(\boldsymbol{\mathcal{X}})$ to obtain $\boldsymbol{\mathcal{A}}_{TT}$. The listing of `FTT-interpolation-construction` is thus omitted. The basis functions (3.8) determine a quadratic convergence of the interpolant to the target function. Thus, for $k > d-1$ and $f \in \mathcal{H}_{\mu}^{k}(\mathbf{I})$:

$$\|f - I_N f_{TT}\|_{L_{\mu}^{2}(\mathbf{I})} \leq \|f\|_{\mathcal{H}_{\mu}^{k}(\mathbf{I})} \sqrt{\frac{(r+1)^{-(k-1)}}{k-1} \frac{r^{d-1}-1}{r-1}} + C N^{-2} |f_{TT}|_{\mathbf{I},\mu,2} \ . \tag{4.38}$$

Additionally these basis functions have local support (as opposed to the global support of the polynomials used for the projection) and this prevents the propagation over all the space of errors due to singularities of f.

The same approach can be taken for the interpolation with Lagrange polynomials. The interpolating values can be obtained extracting the diagonal $f(\mathbf{Y}) \simeq \{(\Pi_{\mathbf{N}} f_{TT})(\boldsymbol{\mathcal{Y}})_{i,\dots,i}\}_{i=1}^{N_y}$ of

$$f(\boldsymbol{\mathcal{Y}}) \simeq (\Pi_{\mathbf{N}} f)(\boldsymbol{\mathcal{Y}}) = \mathbf{L} f(\boldsymbol{\mathcal{X}}), \qquad \mathbf{L} = L^{(1)} \otimes \cdots \otimes L^{(d)}, \tag{4.39}$$

where $L^{(k)}$ is the $N_y \times N_x^{(k)}$ Lagrange interpolation matrix [27]. This interpolation is not carried out directly in high dimension, but we only need to perform one-dimensional interpolations of the

cores (see Alg. 3):

$$
\begin{aligned}
&\left(\Pi_{\mathbf{N}} f_{TT}\right)(\boldsymbol{\mathcal{Y}}) = \mathbf{L} f_{TT}(\boldsymbol{\mathcal{X}}) = \sum_{\alpha_0,\ldots,\alpha_d=1}^{\mathbf{r}} \sigma(\boldsymbol{\alpha})\beta_1(\alpha_0,\mathbf{y}_1,\alpha_1)\cdots\beta_d(\alpha_{d-1},\mathbf{y}_d,\alpha_d)\ , \\
&\beta_n(\alpha_{n-1},\mathbf{y}_n,\alpha_n) = L^{(n)}\gamma_n(\alpha_{n-1},\mathbf{x}_n,\alpha_n)\ .
\end{aligned} \tag{4.40}
$$

The convergence is again dictated by the regularity of the function f. For $k > d-1$ and $f \in \mathcal{H}^k_\mu(\mathbf{I})$:

$$
\|f - \Pi_N f_{TT}\|_{L^2_\mu(\mathbf{I})} \leq \|f\|_{\mathcal{H}^k_\mu(\mathbf{I})} \sqrt{\frac{(r+1)^{-(k-1)}}{k-1}\frac{r^{d-1}-1}{r-1}} + C(k)N^{-k}|f_{TT}|_{\mathbf{I},\mu,k}\ . \tag{4.41}
$$

4.3.3. The algorithm. Suppose we have a function $f : \mathbf{I} \to \mathbb{R}$ where $\mathbf{I} = \times_{i=1}^d I_i$ and $I_i \subseteq \mathbb{R}$, for $i = 1,\ldots,d$. We would like to construct an approximation of f and to evaluate this approximation on an independent set of points $\mathbf{Y}$. The algorithm for *constructing* and *evaluating* the spectral tensor-train approximation of f proceeds as follows:

1. select a suitable set of candidate nodes $\boldsymbol{\mathcal{X}} = \times_{n=1}^d \mathbf{x}_n$ according to the type of approximation to be constructed
2. construct the approximation using Algorithm 1 for the projection approach or directly using `TT-DMRG-cross` [35] on $f(\boldsymbol{\mathcal{X}})$ for the interpolation approach
3. evaluate the the spectral tensor-train approximation on $\mathbf{Y}$ by Algorithm 2 for the projection approach or by Algorithm 3 for the interpolation approach.

In the following we will refer to the `FTT-projection` and the `FTT-interpolation` algorithms as the combination of the two corresponding steps of construction and evaluation.

The practical implementation uses data structures to cache computed values and to store partially computed decompositions. It also fully supports the usage of the Message Passing Interface (MPI) protocol for the parallel evaluation of f during the execution of `TT-DMRG-cross`.

5. Numerical examples. The Spectral tensor-train decomposition is now applied to several high dimensional functions, with the aim of obtaining a surrogate model of it. The quality of such surrogate models will be evaluated using the relative L^2 error[2]:

$$
\frac{\|f - \mathcal{L} f_{TT}\|_{L^2_\mu(\mathbf{I})}}{\|f\|_{L^2_\mu(\mathbf{I})}} = \sqrt{\frac{\int_{\mathbf{I}} (f - \mathcal{L} f_{TT})^2 d\mu}{\int_{\mathbf{I}} f^2 d\mu}}\ , \tag{5.1}
$$

where $\mathcal{L}$ is one of the projection (P_N) or interpolation (I_N, Π_N) operators. This high dimensional integral is estimated using the Monte Carlo estimator, with the number of samples driven by the target relative tolerance of 10^{-2}.

[2] In Sec. 5.1.2 $\|f - \mathcal{L} f_{TT}\|_{L^2_\mu(\mathbf{I})}$ was used in place of (5.1) for consistency with the results obtained in [8].

	f_1	f_2	f_3	f_4	f_5	f_6
b_j	284.6	725.0	185.0	70.3	2040.0	430.0
e_j	1.5	2.0	2.0	1.0	2.0	2.0
a_j	1.5	5.0	1.85	7.03	20.4	4.3

Table 5.1: Normalization parameters for the Genz functions.

5.1. Genz functions. The Genz functions [11, 12] are a set of functions, defined on $[0,1]^d$, frequently used to estimate the properties of approximation schemes. They are defined as follows:

$$
\begin{aligned}
\text{oscillatory}: f_1(\mathbf{x}) &= \cos\left(2\pi w_1 + \sum_{i=1}^{d} c_i x_i\right) \\
\text{product peak}: f_2(\mathbf{x}) &= \prod_{i=1}^{d} \left(c_i^{-2} + (x_i + w_i)^2\right)^{-1} \\
\text{corner peak}: f_3(\mathbf{x}) &= \left(1 + \sum_{i=1}^{d} c_i x_i\right)^{-(d+1)} \\
\text{Gaussian}: f_4(\mathbf{x}) &= \exp\left(-\sum_{i=1}^{d} c_i^2 (x_i - w_i)^2\right) \\
\text{continuous}: f_5(\mathbf{x}) &= \exp\left(-\sum_{i=1}^{d} c_i^2 |x_i - w_i|\right) \\
\text{discontinuous}: f_6(\mathbf{x}) &= \begin{cases} 0 & \text{if any } x_i > w_i \\ \exp\left(\sum_{i=1}^{d} c_i x_i\right) & \text{otherwise} \end{cases}
\end{aligned}
\tag{5.2}
$$

Except for the "discontinuous" function, the parameters $\mathbf{w}$ are drawn uniformly from $[0,1]$. These parameters act as a shift for the function. For the "discontinuous" function $\mathbf{w}$ determines the position of the hyperplane defining a discontinuity of the function. If $\mathbf{w}$ was drawn uniformly also in this case, the probability of being in the non-zero region of the function would decrease exponentially with the dimension. This would make it very hard to obtain an error estimate for our approximation with Monte Carlo method. Then we impose that for $\mathbf{x} \sim \mathcal{U}([0,1]^d)$, $P[\mathbf{x} > \mathbf{w}] = 1/2$. This was achieved selecting $\mathbf{w} \sim \text{Beta}(\alpha, \beta)$, where $\beta = 1$ and $\alpha = \exp\left(\frac{\log(1/2)}{d}\right) / \left(1 - \exp\left(\frac{\log(1/2)}{d}\right)\right)$.

The parameters $\mathbf{c}$ are drawn uniformly from $[0,1]$ and then normalized to $d^{e_j}\|\mathbf{c}\|_1 = b_j$, for j indexing the six Genz functions. The "difficulty" of the function increases monotonically with b_j and e_j is a scaling constant used for different dimensions. The parameters e_j are defined as suggested in [11, 12], while b_j are selected in order to obtain the same test functions used for $d = 10$ in [1]. These are listed in table 5.1.

In order to compare the results obtained by functional tensor-train projection with the Smolyak pseudospectral sparse grid approximation [8], we also consider the normalization $\|\mathbf{c}\|_1 = a_j$ for $d = 5$ with values listed in table 5.1.

The experiments will be performed picking 30 different sets of parameters $\mathbf{w}$ and $\mathbf{c}$ for each Genz function and looking at the L^2 error (5.1) with respect to the number of function evaluations needed to construct an approximation based on the functional tensor-train projection or interpolation, with a desired order or a desired refinement respectively. Both the error estimate and the number of function evaluations can vary depending on the particular function at hand. In particular the number of function evaluations is driven by the procedure for obtaining a pointwise tensor-train approximation on the tensor grid using the `TT-DMRG-cross` algorithm (see sec. 2.3).

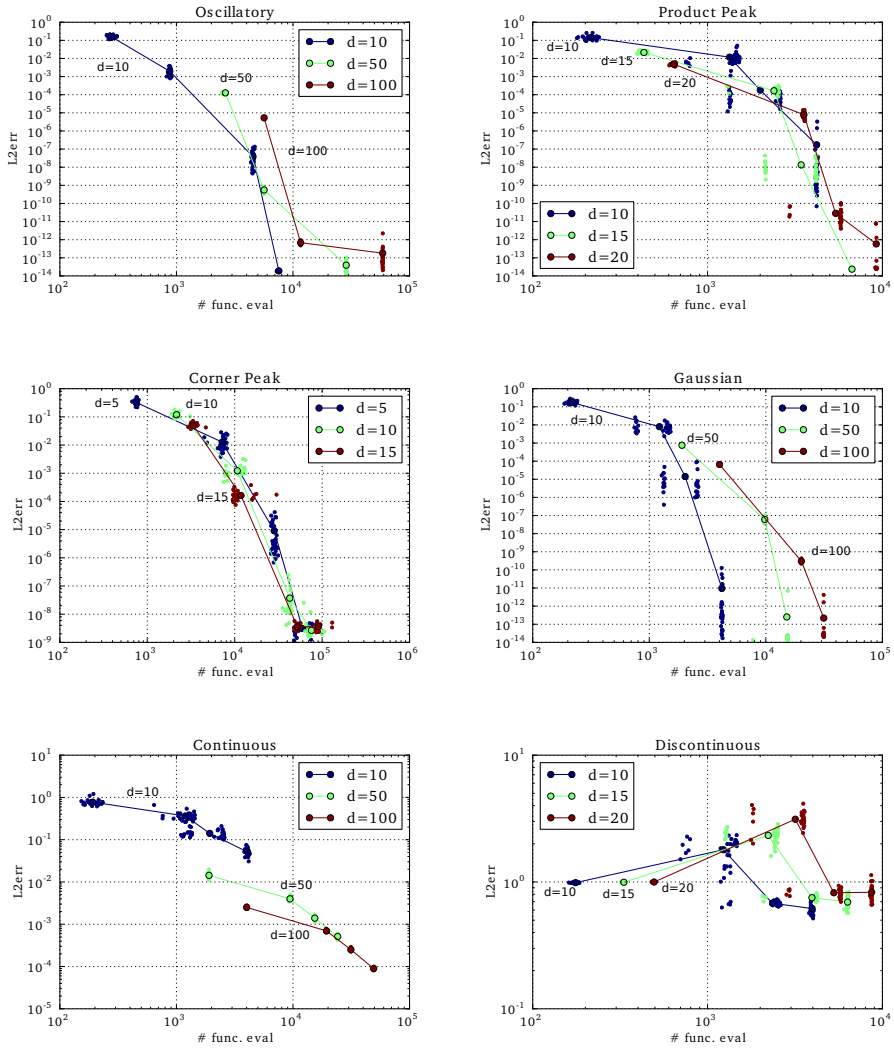


Fig. 5.1: Functional tensor-train projection approximation of the Genz functions. For exponentially increasing polynomial order ($2^i - 1$ for $i = 1, \ldots, 4$) and for different dimensions, 30 Genz functions have been constructed and approximated using the `FTT-projection` algorithm. The scattered dots show the L^2 error and the number of function evaluations needed for each of these realizations. The circled dots represent the mean L^2 error and mean number of function evaluations for increasing polynomial order.

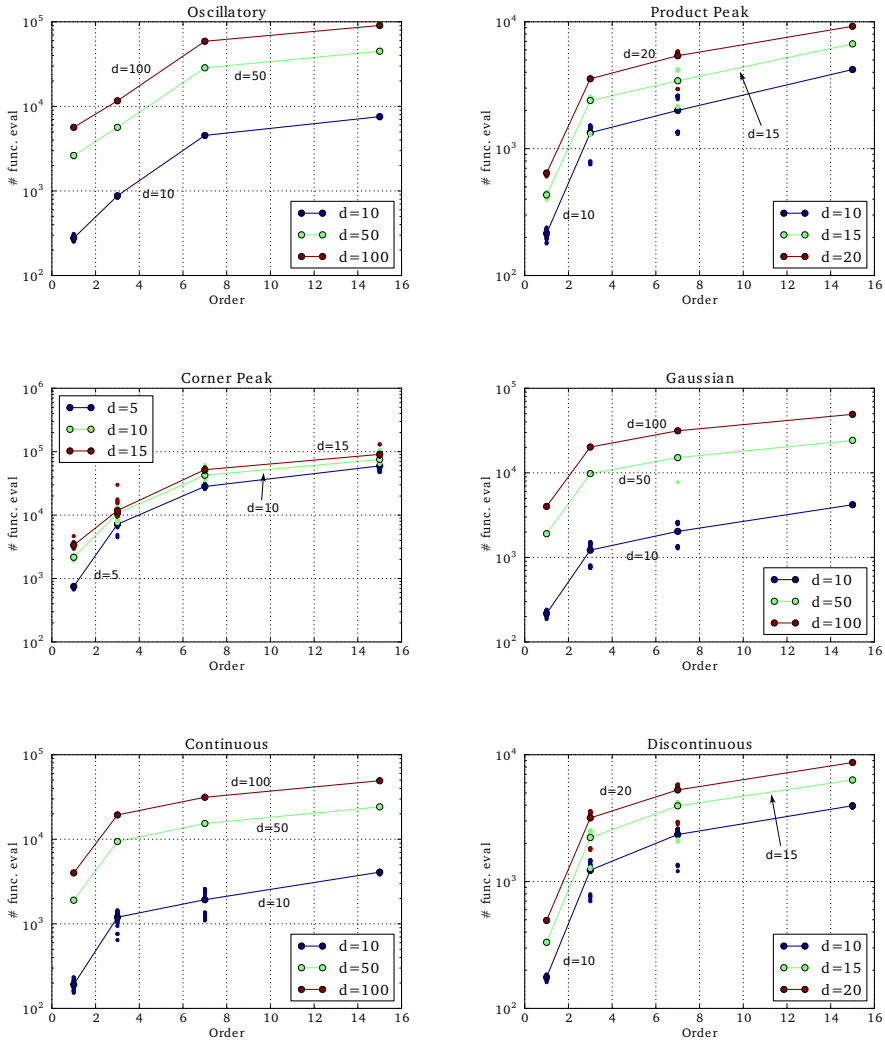


Fig. 5.2: `FTT-projection` approximation of the Genz functions. For exponentially increasing polynomial order ($2^i - 1$ for $i = 1, \ldots, 4$) and for different dimensions, 30 Genz functions have been constructed and approximated using the `FTT-projection` algorithm. The dots show the number of function evaluations with respect to the polynomial order selected.

The target accuracy of the `TT-DMRG-cross` approximation in terms of Frobenious norm is set to $\varepsilon_{\text{rnd}} = 10^{-10}$ to be conservative.

5.1.1. Functional tensor-train projection on the Genz functions. In the next tests different dimensions will be considered ranging between 10 and 100. The "corner peak" function was tested up to $d = 15$ due to the higher computational effort required to build the approximation. The decay of the singular values of this function is very slow, leading to an increased sampling. For the "product peak" function we could not run the tests for $d > 20$ because $f_2 \to 0$ as d increases, leading to a loss of machine precision.

Figure 5.1 shows the convergence rate of the `FTT-projection` approximation on the six Genz functions for exponentially increasing polynomial order ($2^i - 1$ for $i = 1, \dots, 4$). The quadrature points used are Gauss points. Due to the interchangeability of the dimensions in the Genz functions, the single realizations are more scattered for the low-dimensional functions, being these defined by a smaller number of random parameters.

As expected we obtain the *spectral* convergence rate on the smooth functions 1-4. On the "continuous" Genz function the convergence is only quadratic, due to the first order discontinuity in its definition. The approximation to the "discontinuous" function shows no substantial convergence, due to the use of global basis in the approximation of a function with discontinuities. The construction of the approximation of the "corner peak" function requires more function evaluations compared to the other functions: the reason lays in the fact that all the other functions have an exact low rank representation, meaning that the singular values rapidly become zero, leading to no information loss when the truncation is performed in order to select the TT-ranks. The "corner peak" function, instead, couples all the variables with the outer exponentiation, leading to a slower decay of the singular values $\sigma(\boldsymbol{\alpha})$ and to the necessity of increasing the TT-ranks in order to meet the accuracy requirements. The relation between number of function evaluations and the order of the polynomial basis used is shown in figure 5.2. Again, the effect of not being of low-rank, penalizes the performances on the "corner peak" function.

5.1.2. Comparison with sparse grid pseudospectral approximation. The goal of this numerical example is to compare our results to the fully adaptive Smolyak sparse grid pseudospectral approximation [8], where the number of function evaluations is increased by increasing the available computational time. For this test we will consider only the first four smooth Genz functions with $d = 5$ as done in [8]. Figure 5.3 shows that the functional tensor-train projection outperforms the pseudospectral approximation in all the tests where an exact low-rank decomposition of the function exists. The "corner peak" function doesn't have an exact low-rank decomposition and spectral-tensor train is outperformed by the pseudospectral approximation in this case. It is fair to notice however that the fully adaptive pseudospectral approximation performs an anisotropic order adaptation with respect to the dimensions, i.e. it can use different orders on different dimensions. This is a feature that is not yet available in our implementation of spectral tensor-train, thus the increase of order is isotropic, leading to an excessive refinement in certain directions, leaving room for future improvements.

5.2. Modified Genz functions. It is noticeable, from figure 5.1, that the approximations tend to get easier as the dimension is increased. This is due to the fact that the Genz functions were not designed to be used for very high dimensions. As an example, consider the "Gaussian" function f_4. It has the rank one representation:

$$f_4(\mathbf{x}) = \exp\left(-\sum_{i=1}^{d} c_i^2 (x_i - w_i)^2\right) = \prod_{i=1}^{d} \exp\left(-c_i^2 (x_i - w_i)^2\right) \ . \tag{5.3}$$

The $\mathbf{c}$ vector is normalized so that $\|\mathbf{c}\|_1 = \frac{b_j}{d^{e_j}}$. Then, for $d \to \infty$ and for the values of e_j and b_j listed in table 5.1, $c_i \to 0$ and $f_4 \to 1$. This means that the higher is the dimension, the closer the function is to be a constant and thus easier to be approximated.

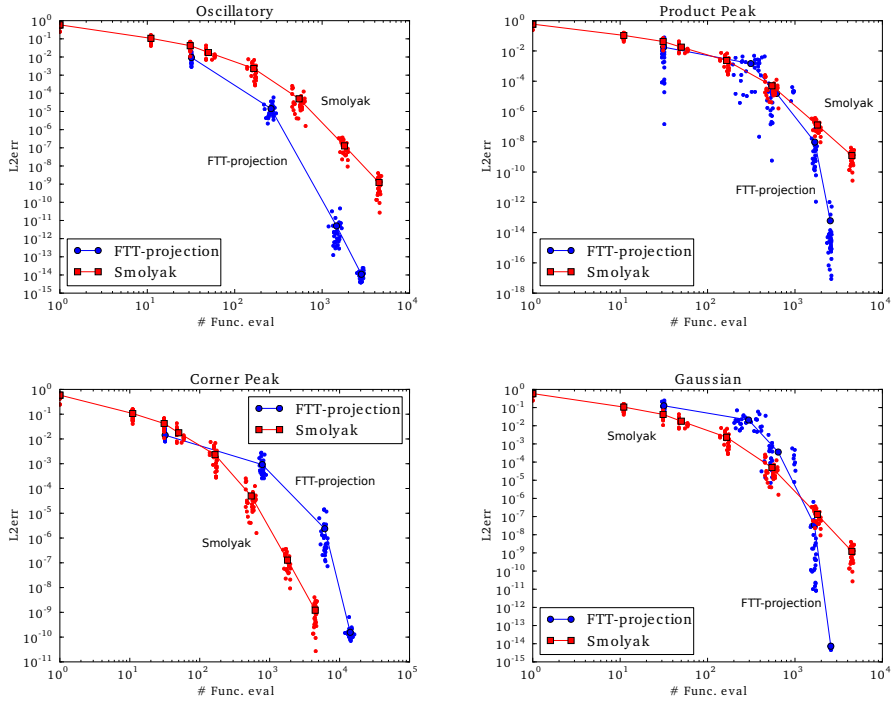


Fig. 5.3: Functional tensor-train projection approximation and Smolyak sparse grid pseudospectral approximation of the Genz functions. For increasing accuracy 30 Genz functions have been constructed and approximated by the two methods. The scattered dots show the L^2 error and the number of function evaluations needed for each of these realizations. The circled/square dots represent the mean L^2 error and mean number of function evaluations for increasing accuracy levels.

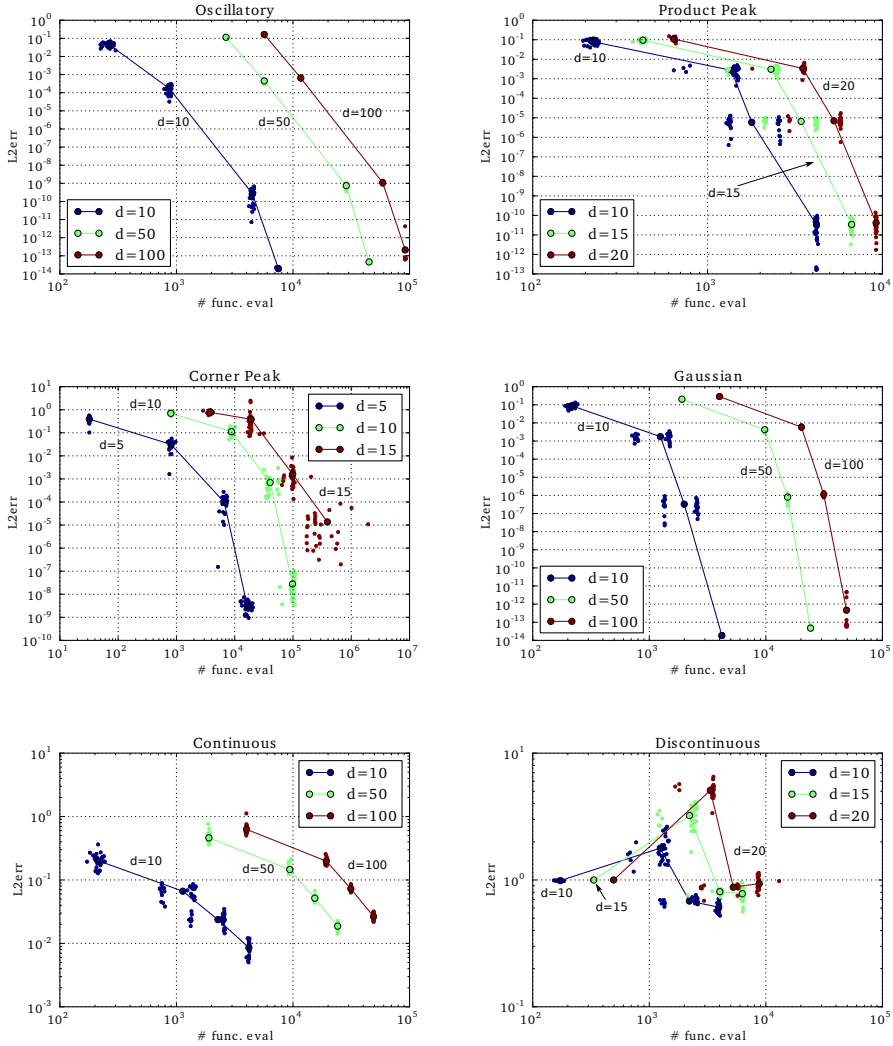


Fig. 5.4: Functional tensor-train projection approximation of the modified Genz functions. For exponentially increasing polynomial order ($2^i - 1$ for $i = 1, \ldots, 4$) and for different dimensions, 30 modified Genz functions have been constructed and approximated using the `FTT-projection` algorithm. The scattered dots show the L^2 error and the number of function evaluations needed for each of these realizations. The circled dots represent the mean L^2 error and mean number of function evaluations for increasing polynomial order.

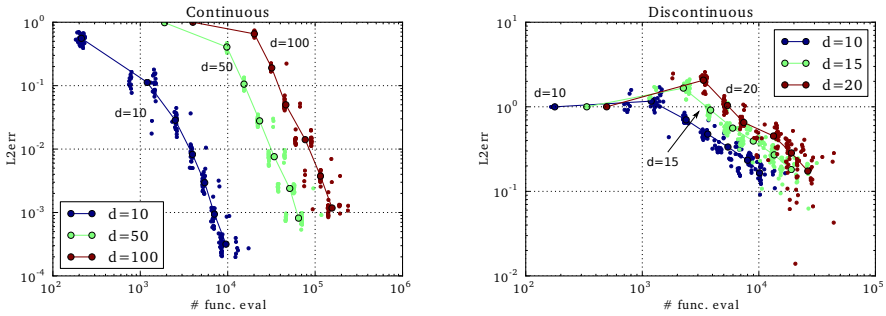


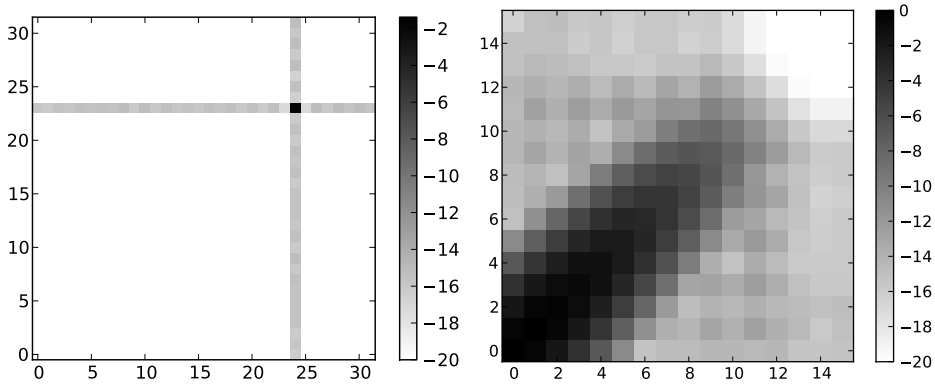
Fig. 5.5: Functional tensor-train linear interpolation of the "continuous" and "discontinuous" modified Genz functions. For exponentially increasing number – from 2^1 to 2^7 – of uniformly distributed interpolating points and for different dimensions, 30 Genz functions have been constructed and approximated by the tensor-train linear interpolation. The scattered dots show the L^2 error and the number of function evaluations needed for each of these realizations. The circled dots represent the mean L^2 error and mean number of function evaluations for increasing grid refinements.

We would instead like to test the performance of the spectral tensor-train approximation on a more realistic set of example functions, whose "difficulty" grows with the dimension. To this end, we use the definition (5.2) of the Genz functions, but we refrain from normalizing the coefficients $\mathbf{c} \sim \mathcal{U}([0,1])$. This leads to functions that don't degenerate to constants in high dimensions, and thus can be used for testing purposes at higher dimensions than the original Genz functions.

5.2.1. Functional tensor-train projection on the modified Genz functions. As a mean of comparison with the original Genz functions, we consider the performances of the functional tensor-train projection on their modified version. Figure 5.4 shows the convergence rate of the surrogate function with respect to the number of function evaluation, for increasing polynomial order. A comparison with figure 5.1 shows that the tests on the Modified Genz functions are more informative about the method with respect to the original functions, because they don't become easier with the increase of dimensions. Again the *spectral* convergence is obtained on the smooth functions. The higher scattering of the points in the approximation of the "corner peak" function is due to the absence of an analytic low-rank representation for such function, and thus the introduction of truncation in the tensor-train decomposition.

5.2.2. Functional tensor-train interpolation on the modified Genz functions. The linear `FTT-interpolation` has been tested on all the modified Genz functions, with an exponentially increasing number – from 2^1 to 2^7 – of uniformly distributed points. Due to space constraints, figure 5.5 shows the convergence rates only for the "continuous" and "discontinuous" Genz functions. For the first four smooth functions we experience at least second order convergence rates, as expected from the choice of linear basis functions. The convergence of the approximation to the "continuous" function is second order, while the convergence rate on the "discontinuous" function is almost first order. This type of convergence is obtained thanks to the locality of the selected basis functions which prevent the error caused by the unresolved discontinuity from globally corrupting the approximation.

The Lagrange `FTT-interpolation` has also been tested for all the modified Genz functions. We omit the results here because they are in line with the results obtained with the `FTT-projection` showed in Fig. 5.1.



(a) f_1: $d = 2$, Rank=1, #f.eval. $209/32^2$
$d = 3$, Rank=1, #f.eval. $626/32^3$
$d = 4$, Rank=1, #f.eval. $1210/32^4$
$d = 5$, Rank=1, #f.eval. $1442/32^5$

(b) f_2: $d = 2$, Rank=11, #f.eval. $256/16^2$
$d = 5$, $J = [1, 2]$, #f.eval. $3935/16^5$
$d = 5$, $J = [0, 4]$, #f.eval. $73307/16^5$

Fig. 5.6: Magnitude of the Fourier coefficients, in $\log_{10}$ scale, for functions f_1 and f_2, obtained using the `TT-projection` algorithm to a precision of $\varepsilon = 10^{-10}$. The corresponding maximum TT-rank and the number of function evaluations with respect to the total grid size are listed for several dimensions.

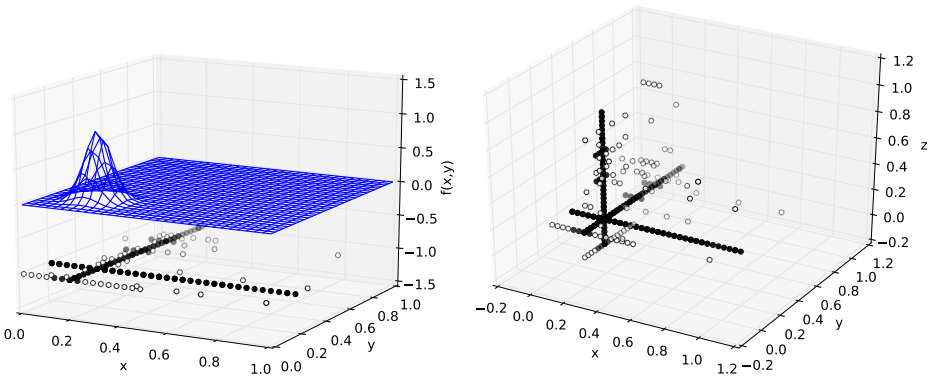
5.3. `FTT-projection` and mixed Fourier modes. It is now understood that the approximation of multidimensional functions with sparse grids is exact when the function's Fourier coefficients are non-zero only for the set of admissible multi-indices included in the sparse grid construction [8, 9]. The convergence of the approximation deteriorates when the decay of the Fourier coefficients is slow for mixed modes.

We construct two ad-hoc functions to highlight some properties of the `FTT-projection`, when approximating functions with different types of decay in their Fourier coefficients. Let us consider functions defined on $\mathbf{I} = I_1 \times \cdots \times I_d$ where $I_i = [-1, 1]$. On this hypercube we consider the sub-cube $I_{j_1} \times \cdots \times I_{j_c}$, where $J = \{j_i\}_{i=1}^c \subseteq [1, \ldots, d]$. For every index in J, we select $\{n_{j_i}\}_{i=1}^c > 0$ to be the maximum order of polynomials included in the functions along the i-th direction. The functions will then be defined as follows:

$$
\begin{aligned}
f_1(\mathbf{x}) &= \prod_{k=1}^{c} \phi_{l_k}(x_{j_k}) \,, \\
f_2(\mathbf{x}) &= \sum_{i_{j_1}=0}^{n_{j_1}} \cdots \sum_{i_{j_c}=0}^{n_{j_c}} \left[\exp\left(-\mathbf{i}^T \boldsymbol{\Sigma} \mathbf{i}\right) \prod_{k=1}^{c} \phi_{i_{j_k}}(x_{j_k}) \right] \,,
\end{aligned}
\tag{5.4}
$$

where $\boldsymbol{\Sigma}$ is a $c \times c$ matrix defining the level of interaction between different dimensions, $\{\phi_{i_{j_k}}\}_{i_{j_k}=1}^{n_{j_k}}$ are chosen to be the normalized Legendre polynomials, $\mathbf{i} = (i_{j_1}, \ldots, i_{j_c})^T$ and the ϕ_{l_k} are possibly high order polynomials. To simplify the notation, we will set $n_{j_k} = n$ for all j_k.

The function f_1 is a function with one single high mixed Fourier mode as shown in figure 5.6a. In spite of the high polynomial order, the rank of the function is correctly estimated to be 1 and thus very few sampling points are needed in order to recover the required precision. This highlights that, on the contrary of sparse grids, the spectral tensor-train does not discard basis functions in its construction, but it uses always a fully tensorized set of basis functions.



(a) $d = 2$, #f.eval. $119/32^2$ **(b)** $d = 3$, #f.eval. $214/32^3$

Fig. 5.7: The left figure shows the off-centered local feature of (5.5). The white and black dots show the candidate points where the function has been evaluated. The black dots show the points that are used in the final `TT-DMRG-cross` approximation. `TT-DMRG-cross` detects the feature and clusters the nodes around it, in order to obtain maximum accuracy ($\varepsilon = 10^{-10}$). The right figure shows the same test for $d = 3$.

The function f_2 aims to represent a function with a slow decay of mixed Fourier coefficients in the J dimensions. The function is constant along the remaining dimensions. For $d = 2$ we set

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & -0.9 \\ -0.9 & 1 \end{bmatrix}$$

and $J = [0, 1]$. The decay of the coefficients, as estimated using the `FTT-projection`, is shown in figure 5.6b. The function has an high TT-rank and this leads to the complete sampling of the space. We can use this function also to experiment on what is called the *ordering problem* of the TT-decomposition. We let $d = 5$ and use different combinations of indices in J. If J contains two neighboring dimensions, $J = [1, 2]$ in the example, the TT-ranks of the decomposition will be $\mathbf{r} = [1, 1, 11, 1, 1, 1]$, where the maximum is attained between the cores G_1 and G_2. If we consider J containing non-neighboring dimensions, $J = [0, 4]$ in the example, we practically obtain the same function, with reordered dimensions. In this case the TT-ranks will be $\mathbf{r} = [1, 11, 11, 11, 11, 1]$. This happens due to the hierarchical construction of the TT-decomposition, where information can be propagated only from one core to the next one. The example shows that the only consequence of a wrong ordering choice is that it can lead to an increased number of function evaluations, which grows with r^2. This however does not affect the accuracy of the approximation.

5.4. Resolution of local features. It is often the case that the modeled function presents important local features which need to be resolved accurately. An a priori clustering of nodes is not possible because the location of such feature is unknown. The `TT-DMRG-cross` algorithm overcomes this problem, because it adaptively selects the nodes that are relevant for the approximation, thus exploring the space with an increasing knowledge about the features of the function. As an explanatory example, consider

$$f(\mathbf{x}) = \exp\left(-\frac{|\mathbf{x} - \mathbf{x}_0|^2}{2l^2}\right) \quad . \tag{5.5}$$

Let $d = 2$, $\mathbf{x}_0 = [0.2, 0.2]$ and $l = 0.05$. The function shows an off-centered peak as shown in figure 5.7a. The points used by `TT-DMRG-cross` (with accuracy $\varepsilon = 10^{-10}$) are shown on the same figure,

where the white dots are the points used on the way to the final approximation, while the black dots are the points retained in the final approximation. Figure 5.7b shows the set of points used for $d = 3$ and $\mathbf{x}_0 = [0.2, 0.2, 0.2]$. The same kind of clustering is observed.

5.5. Elliptic equation with random input data. Here we consider the classical Poisson's equation defined on the unit square $\Gamma = [0, 1] \times [0, 1]$

$$\begin{cases} -\nabla \cdot (\kappa(\mathbf{x}, \omega)\nabla u(\mathbf{x}, \omega)) = f(\mathbf{x}, \omega) & \text{in} \quad \Gamma \times \Omega \\ u(\mathbf{x}, \omega) = 0 & \text{on} \quad \partial\Gamma \times \Omega \end{cases} , \tag{5.6}$$

where $f(\mathbf{x}, \omega) = 1$ is a deterministic load, and κ is a log-normal random field defined on the probability space (Ω, Σ, μ) by

$$\kappa(\mathbf{x}, \omega) = \exp\left(\frac{g(\mathbf{x}, \omega)}{10}\right) , \qquad g(\mathbf{x}, \omega) \sim \mathcal{N}\left(\mathbf{0}, C_g(\mathbf{x}, \mathbf{y})\right) . \tag{5.7}$$

We characterize the normal random field $g \in L^2_\mu(\Omega; L^\infty(\Gamma))$ by the squared exponential covariance:

$$C_g(\mathbf{x}, \mathbf{y}) = \int_\Omega g(\mathbf{x}, \omega) g(\mathbf{y}, \omega) d\mu(\omega) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2l^2}\right) , \tag{5.8}$$

where $l > 0$ determines the spatial correlation length of the field. We decompose the random field through the Karhunen-Loève (KL) expansion [30]

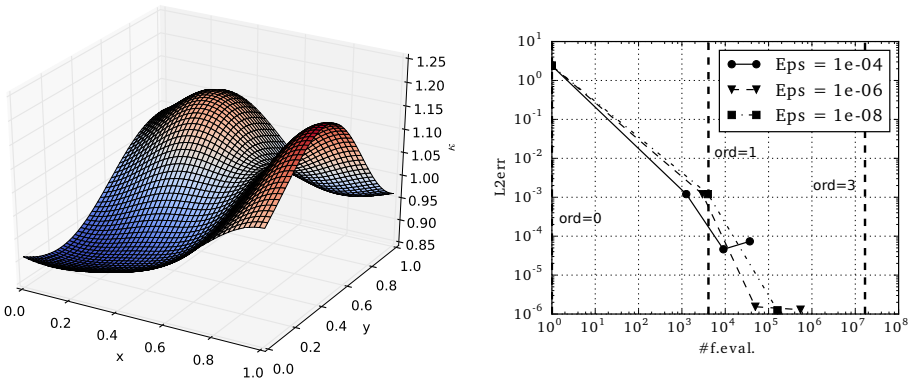
$$g(\mathbf{x}, \omega) = \sum_{i=1}^{\infty} \sqrt{\lambda_i} \chi_i(\mathbf{x}) Y_i(\omega) , \tag{5.9}$$

where $Y_i \sim \mathcal{N}(0, 1)$ and $\{\lambda_i, \chi_i(\mathbf{x})\}_{i=1}^{\infty}$ are the eigenvalues and eigenfunctions of the eigenvalue problem $\int_{\mathbf{y}\in\Gamma} C_g(\mathbf{x}, \mathbf{y})\chi_i(\mathbf{y})d\mathbf{y} = \lambda_i \chi_i(\mathbf{x})$. The KL-expansion is truncated in order to retain the 95% of the total variance ($\mathrm{Var}[g(\mathbf{x}, \omega)] = \mathbf{1}$), i.e. we find $d \in \mathbb{N}^+$ such that $\sum_{i=1}^{d} \lambda_i \geq 0.95$. With a correlation length of $l = 0.25$ we use $d = 12$ terms in the KL-expansion. Figure 5.8a shows one realisation of the random field (5.7), computed using the selected parameters for the KL-expansion (5.9). The use of the KL-expansion allows (5.6) to be turned into a parametric problem, where we seek the solution $u \in L^2(\Gamma) \times L^2_{d_\mathbf{Y}}(\mathbb{R}^d)$. Here we will focus on the construction of a surrogate of $u(\mathbf{x}_0, \mathbf{Y})$, for $\mathbf{x}_0 = (0.75, 0.25)$.

The surrogate is constructed using the `FTT-projection` with Hermite polynomials as basis functions. Figure 5.8b shows the convergence, in terms of the L^2 error (5.1), for orders 0, 1, 3 and 7 and for different target accuracies. These accuracies are driven by the tolerances that are set in the `TT-DMRG-cross` algorithm, and they represent the accuracy with which the discrete tensor of values is approximated by the TT-decomposition. We can see that the accuracy of the surrogate improves spectrally until the target accuracy is reached. After this happens, an increase in the order of the surrogate doesn't provide any more improvement and the convergence plot flattens at the target accuracy level.

6. Conclusions. This paper presents a novel and rigorous construction of the Spectral tensor-train decomposition, that can be used for the approximation of high-dimensional functions. The method aims at tackling the *curse of dimensionality* for functions with sufficient regularity, exploiting the low-rank representation of the approximated function, and at attaining *spectral convergence*, by the use of polynomial approximation.

We present an iterative procedure to decompose an arbitrary function $f \in L^2_\mu(\mathbf{I})$, obtaining a format that we call the functional tensor-train decomposition, to distinguish it from the already studied discrete tensor-train decomposition. The construction of the surrogate is based on the existence of the singular value decomposition of Hilbert-Schmidt kernels in $L^2_\mu(\mathbf{I})$ and on the regularity



(a) KL-expansion **(b)** Convergence

Fig. 5.8: The left figure shows a realization of the random field (5.7), evaluated using the truncated KL-expansion (5.9). The right figure shows the convergence of the `FTT-projection` of orders 0, 1, 3 and 7 for different target accuracies selected. The vertical dashed lines show the number of function evaluations that would be required to attain a full tensor approximation.

properties of the function (c.f. Thm. 4.7). This regularity will be carried on by the singular functions of the decomposition (c.f. Thm. 4.9 and 4.11), leading to the same convergence rate that would be obtained if we applied the polynomial approximation to the high-dimensional f.

The tensor-train decomposition [31] obtained through the `TT-DMRG-cross` algorithm [35] leads to a memory and computational complexity that scale linearly with the dimensionality of the function. The theory of polynomial approximation is added on top of the discrete representation obtained by `TT-DMRG-cross`, and provides an accurate approximation that converges *spectrally* on smooth functions. The user is required to select the polynomial order of the approximation and the overall accuracy required. The latter tolerance will drive the amount of dimensional interaction described by the approximation and ultimately the number of function evaluations, which will grow mildly for functions with a fast decay of their singular values.

Unlike in sparse grid pseudospectral approximation, the method doesn't make any *a priori* assumption in the choice of the basis for the separation of the space $L^2_\mu(\mathbf{I})$. Instead, it uses the singular functions of f, which are optimal. The choice of a polynomial basis is made during the projection of the singular functions $\gamma_k(\alpha_{k-1}, \cdot, \alpha_k) \in L^2_\mu(I_k)$ onto the space spanned by such fully tensorized polynomials. This approach also permits to resolve local features not positioned at the center of the domain, by clustering the evaluation points close to the feature.

In some cases, the performances of the method are dependent on the ordering of the dimensions. This results only in a higher number of function evaluations, although still linear in d, but does not compromises the quality of the approximation. Research in the direction of finding an optimal ordering *a priori* is a topic of ongoing work.

The results from this work pave the way to an adaptive spectral tensor-train decomposition: the smoothness properties of the singular functions can in fact be used as an indicator for the necessity of increasing the polynomial order on each dimension. This will allow the complete automation of the construction of spectral tensor-train surrogates.

The results in this work have been obtained using the open-source Python library for Spectral

tensor-train decomposition that is made available on-line[3] – including examples from this paper.

Acknowledgments. The authors would like to thank Jan S. Hesthaven, Alessio Spantini, Florian Augustin and Patrick Conrad for fruitful discussions on this topic and for providing many useful comments on the paper. We would also like to thank Dr. Dmitry Savostyanov who called our attention to the `TT-DMRG-cross` algorithm, just after reading a preprint of this work.

Appendix A. Hölder continuity and the Smithies' condition. In Section 4.2 we use a result by Smithies [40, Thm. 14] to prove the boundedness of the weak derivatives of the cores of the FTT-decomposition. The condition under which Smithies' result hold is:

DEFINITION A.1 (Smithies' integrated Hölder continuity). *Let $K(s,t)$ defined for $s,t \in [a,b]$. Without loss of generality, let $a=0$ and $b=\pi$. For $r>0$, let*

$$K^{(i)}(s,t) = \frac{\partial^i K(s,t)}{\partial s^i}\,, \qquad 0 < i \leq r \tag{A.1}$$

and let $K^{(1)},\dots,K^{(r-1)}$ exist and continuous. Let $K^{(r)} \in L^p(s)$ a.e. in t and $1<p\leq 2$. The integrated Hölder continuity, with either $r>0$ and $\alpha>0$ or $r=0$ and $\alpha > \frac{1}{p}-\frac{1}{2}$, holds for K if there exists $A>0$ such that:

$$\int_0^\pi \left\{ \int_0^\pi \left| K^{(r)}(s+\theta,t) - K^{(r)}(s-\theta,t) \right|^p ds \right\}^{\frac{2}{p}} dt \leq A|\theta|^{2\alpha}\,. \tag{A.2}$$

This definition is of difficult interpretation. Furthermore, in the scope of this work, we are interested in the case $r=0$. A simpler, but not equivalent, definition is the one mentioned in [41]:

DEFINITION A.2 (Hölder continuity almost everywhere). *Let $K(s,t)$ defined for $s,t\in[a,b]$. K is Hölder continuous a.e. with exponent $\alpha>0$ if there exists $C>0$ such that*

$$|K(s+\theta,t) - K(s-\theta,t)| \leq C|\theta|^\alpha \tag{A.3}$$

almost everywhere in t.

For the sake of simplicity, we show that:

PROPOSITION A.3. *The Hölder continuity a.e. is a sufficient condition for the Smithies' integrated Hölder continuity.*

Proof. Let $K \in L^p(s)$ for almost all t, $1<p\leq 2$. For $\alpha > \frac{1}{2}$, let K be Hölder continuous a.e. in t. Then:

$$\begin{aligned}\int_0^\pi \left\{ \int_0^\pi \left| K^{(r)}(s+\theta,t) - K^{(r)}(s-\theta,t) \right|^p ds \right\}^{\frac{2}{p}} dt &\leq \int_0^\pi \left\{ \int_0^\pi C^p\,|\theta|^{\alpha p}\, ds \right\}^{\frac{2}{p}} dt \\ &= C^2 \pi^{\frac{3}{p}}\,|\theta|^{2\alpha} \leq C^2\pi^3\,|\theta|^{2\alpha} = A\,|\theta|^{2\alpha}\end{aligned} \tag{A.4}$$

where we recognize the bound (A.2) of the Smithies' integrated Hölder continuity. ∎

Appendix B. Proves of auxiliary results for theorem 4.7.

Proof. [Proof of lemma 4.5] By definition of Sobolev norm, seminorm and weak derivative $D^{\mathbf{i}}$:

$$\begin{aligned}|J|^2_{I_1\times I_1,\mu,k} \leq \|J\|^2_{\mathcal{H}^k_\mu(I_1\times I_1)} &= \sum_{|\mathbf{i}|=0}^{k} \|D^{\mathbf{i}}\langle f(x,y), f(\bar{x},y)\rangle_{L^2_\mu(\bar{\mathbf{I}})}\|^2_{L^2_\mu(I_1\times I_1)} \\ &= \sum_{|\mathbf{i}|=0}^{k} \|\langle D^{i_1,\mathbf{0}} f(x,y), D^{i_2,\mathbf{0}} f(\bar{x},y)\rangle_{L^2_\mu(\bar{\mathbf{I}})}\|^2_{L^2_\mu(I_1\times I_1)}\,,\end{aligned} \tag{B.1}$$

[3] https://pypi.python.org/pypi/TensorToolbox/

where $\mathbf{i}$ is a two dimensional multi-index. Using the Cauchy-Schwarz inequality, it holds:

$$\|\langle D^{i_1,\mathbf{0}} f(x,y), D^{i_2,\mathbf{0}} f(\bar{x},y)\rangle_{L^2_\mu(\bar{\mathbf{I}})}\|^2_{L^2_\mu(I_1\times I_1)} \leq \|D^{i_1,\mathbf{0}} f(x,y)\|^2_{L^2_\mu(\mathbf{I})}\|D^{i_2,\mathbf{0}} f(x,y)\|^2_{L^2_\mu(\mathbf{I})} \tag{B.2}$$

Let now $\mathbf{j}$ and $\mathbf{l}$ be two d-dimensional multi-indices, then (B.1) can be bounded by

$$\begin{aligned} |J|^2_{I_1\times I_1,\mu,k} \leq \|J\|^2_{\mathcal{H}^k_\mu(I_1\times I_1)} &\leq \sum_{|\mathbf{i}|=0}^{k} \|D^{i_1,\mathbf{0}} f(x,y)\|^2_{L^2_\mu(\mathbf{I})}\|D^{i_2,\mathbf{0}} f(x,y)\|^2_{L^2_\mu(\mathbf{I})} \\ &\leq \sum_{|\mathbf{j}|=0}^{k}\sum_{|\mathbf{l}|=0}^{k} \|D^{\mathbf{j}} f(x,y)\|^2_{L^2_\mu(\mathbf{I})}\|D^{\mathbf{l}} f(x,y)\|^2_{L^2_\mu(\mathbf{I})} \leq \|f\|^4_{\mathcal{H}^k_\mu(\mathbf{I})} \,. \end{aligned} \tag{B.3}$$

Since $\|J\|_{\mathcal{H}^k_\mu(I_1\times I_1)} \leq \|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} < \infty$ by assumption, then $J \in \mathcal{H}^k_\mu(I_1\times I_1)$. □

Proof. [Proof of lemma 4.6] Using the definition of Sobolev norm and theorem 4.11,

$$\begin{aligned} \|\phi_1(i_1)\|^2_{\mathcal{H}^k_\mu(\bar{\mathbf{I}})} &= \sum_{|\mathbf{j}|=0}^{k} \|D^{\mathbf{j}}\phi_1(i_1)\|^2_{L^2_\mu(\bar{\mathbf{I}})} \leq \sum_{|\mathbf{j}|=0}^{k} \frac{1}{\lambda(i_1)}\|\psi_1(i_1)\|^2_{L^2_\mu(I_1)}\|D^{0,\mathbf{j}} f\|^2_{L^2_\mu(\mathbf{I})} \\ &= \frac{1}{\lambda(i_1)}\sum_{|\mathbf{j}|=0}^{k} \|D^{0,\mathbf{j}} f\|^2_{L^2_\mu(\mathbf{I})} \leq \frac{1}{\lambda(i_1)}\sum_{|\mathbf{l}|=0}^{k} \|D^{\mathbf{l}} f\|^2_{L^2_\mu(\mathbf{I})} = \frac{1}{\lambda(i_1)}\|f\|^2_{\mathcal{H}^k_\mu(\mathbf{I})} \,. \end{aligned} \tag{B.4}$$

□

REFERENCES

[1] Volker Barthelmann, Erich Novak, and Klaus Ritter, *High dimensional polynomial interpolation on sparse grids*, Advances in Computational Mathematics, 12 (2000), pp. 273–288.

[2] C Bernardi and Y Maday, *Polynomial interpolation results in Sobolev spaces*, Journal of computational and applied mathematics, (1992).

[3] Susanne C. Brenner and L. Ridgway Scott, *The Mathematical Theory of Finite Element Methods*, vol. 15 of Texts in Applied Mathematics, Springer New York, New York, NY, 2008.

[4] R. Bro, *Multi-way analysis in the food industry: models, algorithms, and applications*, PhD thesis, Universiteit van Amsterdam, 1998.

[5] C. Canuto, M.Y. Hussaini, A. Quarteroni, and T.A. Zang, *Spectral Methods - Fundamentals in Single Domains*, Scientific Computation, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[6] JD Carroll and JJ Chang, *Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition*, Psychometrika, 35 (1970).

[7] Ali Çivril and Malik Magdon-Ismail, *On selecting a maximum volume sub-matrix of a matrix and related problems*, Theoretical Computer Science, 410 (2009), pp. 4801–4811.

[8] PR Conrad and YM Marzouk, *Adaptive Smolyak pseudospectral approximations*, SIAM Journal on Scientific Computing, (2013).

[9] Paul G. Constantine, Michael S. Eldred, and Eric T. Phipps, *Sparse pseudospectral approximation method*, Computer Methods in Applied Mechanics and Engineering, 229-232 (2012), pp. 1–12.

[10] Walter Gautschi, *Orthogonal Polynomials: Computation and Approximation*, Numerical Mathematics and Scientific Computation, Oxford University Press, 2004.

[11] A Genz, *Testing multidimensional integration routines*, Proc. of international conference on Tools, methods and languages for scientific and engineering computation, (1984).

[12] ———, *A package for testing multiple integration subroutines*, Numerical Integration, (1987).

[13] A. Genz and B. D. Keister, *Fully symmetric interpolatory rules for multiple integrals over infinite regions with Gaussian weight*, Journal of Computational and Applied Mathematics, 71 (1996), pp. 299–309.

[14] GH Golub and JH Welsch, *Calculation of Gauss quadrature rules*, Mathematics of Computation, (1969), pp. 221–230.

[15] S Goreinov and I Oseledets, *How to find a good submatrix*, in Matrix methods: Theory, Algorithms and Applications, World Scientific Publishing, Co. Pte., Ltd., Singapore, 2010, pp. 247–256.

[16] SA Goreinov, NL Zamarashkin, and EE Tyrtyshnikov, *Pseudo-skeleton approximations by matrices of maximal volume*, Mathematical Notes, 62 (1997), pp. 619–623.

[17] Lars Grasedyck, *Hierarchical singular value decomposition of tensors*, SIAM Journal on Matrix Analysis and Applications, 31 (2010), pp. 2029–2054.
[18] Lars Grasedyck, Daniel Kressner, and Christine Tobler, *A literature survey of low-rank tensor approximation techniques*, arXiv preprint arXiv:1302.7121, (2013), pp. 1–20.
[19] P R Halmos and V S Sunder, *Bounded integral operators on L2 spaces*, Ergebnisse der Mathematik und ihrer Grenzgebiete, Springer-Verlag, 1978.
[20] A. Hammerstein, *Über die Entwickelung des Kernes linearer Integralgleichu ngen nach Eigenfunktionen*, Sitzungsberichte Preuss. Akad. Wiss., (1923), pp. 181–184.
[21] G. H. Hardy and J. E. Littlewood, *Some new properties of fourier constants*, Mathematische Annalen, 97 (1927), pp. 159–209.
[22] RA Harshman, *Foundations of the PARAFAC procedure: models and conditions for an" explanatory" multimodal factor analysis*, UCLA Working Papers in Phonetics, (1970), pp. 1–84.
[23] Jan S. Hesthaven and Tim Warburton, *Nodal Discontinuous Galerkin Methods*, vol. 54 of Texts in Applied Mathematics, Springer New York, New York, NY, 2008.
[24] K Jörgens, *Linear integral operators*, Surveys and reference works in mathematics, Pitman Advanced Pub. Program, 1982.
[25] BN Khoromskij, *O (dlog N)-Quantics Approximation of Nd Tensors in High-Dimensional Numerical Modeling*, Constructive Approximation, (2011), pp. 257–280.
[26] Tamara G. Kolda and Brett W. Bader, *Tensor Decompositions and Applications*, SIAM Review, 51 (2009), pp. 455–500.
[27] David A. Kopriva, *Implementing Spectral Methods for Partial Differential Equations*, Scientific Computation, Springer Netherlands, Dordrecht, 2009.
[28] E Kreyszig, *Introductory functional analysis with applications*, Wiley classics library, John Wiley & Sons, 2007.
[29] JB Kruskal, RA Harshman, and ME Lundy, *How 3-MFA data can cause degenerate PARAFAC solutions, among other relationships*, Multiway data analysis, (1989).
[30] M. Loève, *Probability Theory*, vol. I-II, Springer-Verlang, New York, 4 ed., 1978.
[31] IV Oseledets, *Tensor-train decomposition*, SIAM Journal on Scientific Computing, 33 (2011), pp. 2295–2317.
[32] Ivan Oseledets and Eugene Tyrtyshnikov, *TT-cross approximation for multidimensional arrays*, Linear Algebra and its Applications, 432 (2010), pp. 70–88.
[33] I V Oseledets, *Approximation of $2^d times 2^d$ Matrices Using Tensor Decomposition*, SIAM Journal on Matrix Analysis and Applications, 31 (2010), pp. 2130–2145.
[34] Knut Petras, *Smolyak cubature of given polynomial degree with few nodes for increasing dimension*, Numerische Mathematik, 93 (2003), pp. 729–753.
[35] Dmitry Savostyanov and Ivan Oseledets, *Fast adaptive interpolation of multi-dimensional arrays in tensor train format*, The 2011 International Workshop on Multidimensional (nD) Systems, (2011), pp. 1–8.
[36] E Schmidt, *Zur Theorie der linearen und nicht linearen Integralgleichungen Zweite Abhandlung*, Mathematische Annalen, 63 (1907), pp. 433–476.
[37] Christoph Schwab and Radu Alexandru Todor, *Karhunen-Loève approximation of random fields by generalized fast multipole methods*, Journal of Computational Physics, 217 (2006), pp. 100–122.
[38] ND Sidiropoulos and Rasmus Bro, *On the uniqueness of multilinear decomposition of N-way arrays*, Journal of chemometrics, (2000), pp. 229–239.
[39] V De Silva and LH Lim, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM Journal on Matrix Analysis and Applications, (2008), pp. 1–44.
[40] F Smithies, *The eigen-values and singular values of integral equations*, Proceedings of the London Mathematical Society, (1937).
[41] A Townsend and LN Trefethen, *Continuous analogues of matrix factorizations*, people.maths.ox.ac.uk, (2013), pp. 1–22.
[42] L N Trefethen and D Bau III, *Numerical linear algebra*, vol. 12, Society for Industrial and Applied Mathematics, 1997.
[43] LR Tucker, *Implications of factor analysis of three-way matrices for measurement of change*, in Problems in measuring change, C W Harris, ed., University of Wisconsin Press, Madison WI, 1963, pp. 122–137.
[44] J Šimša, *The bestL 2-approximation by finite sums of functions with separable variables*, aequationes mathematicae, 43 (1992), pp. 248–263.
[45] SR White, *Density-matrix algorithms for quantum renormalization groups*, Physical Review B, 48 (1993), pp. 345–356.

Bibliography

[1] D. Bigoni, A. P. Engsig-Karup, and H. True. "Comparison of Classical and Modern Uncertainty Quantification Methods for the Calculation of Critical Speeds in Railway Vehicle Dynamics". In: *13th mini Conference on Vehicle System Dynamics, Identification and Anomalies*. Budapest, Hungary, 2012.

[2] D. Bigoni, A. P. Engsig-Karup, and H. True. "Anwendung der Uncertainty Quantification bei eisenbahndynamischen problemen". In: *Z E Vrail - Glasers Annalen* 137.SPL.ISSUE (2013), pp. 152–158. ISSN: 1618-8330.

[3] D. Bigoni, A. P. Engsig-Karup, and H. True. "Modern Uncertainty Quantification Methods in Railroad Vehicle Dynamics". In: *ASME 2013 Rail Transportation Division Fall Technical Conference*. Altoona: ASME, Oct. 2013, V001T01A009. ISBN: 978-0-7918-5611-6. DOI: 10.1115/RTDF2013-4713.

[4] D. Bigoni, H. True, and A. P. Engsig-Karup. "Sensitivity Analysis of the critical speed in railway vehicle dynamics". In: *23rd IAVSD Symposium on Dynamics of Vehicles on Roads and Tracks*. step C. Qingdao, 2013.

[5] D. Bigoni, H. True, and A. P. Engsig-Karup. "Sensitivity analysis of the critical speed in railway vehicle dynamics". In: *Vehicle System Dynamics* May 2014 (Apr. 2014), pp. 272–286. ISSN: 0042-3114. DOI: 10.1080/00423114.2014.898776.

[6] D. Bigoni, A. P. Engisg-Karup, and H. True. "Global Sensitivity Analysis of Railway Vehicle Dynamics on Curved Tracks". In: *Volume 2: Dynamics, Vibration and Control; Energy; Fluids Engineering; Micro and Nano Manufacturing*. Copenhagen, Denmark: ASME, July 2014, V002T07A023. ISBN: 978-0-7918-4584-4. DOI: 10.1115/ESDA2014-20529.

[7] H. True, A. Engsig-Karup, and D. Bigoni. "On the numerical and computational aspects of non-smoothnesses that occur in railway vehicle dynamics". In: *Mathematics and Computers in Simulation* 95 (Jan. 2014), pp. 78–97. ISSN: 03784754. DOI: 10.1016/j.matcom.2012.09.016.

[8] D. Bigoni, A. P. Engsig-Karup, and C. Eskilsson. "A Stochastic Nonlinear Water Wave Model for Efficient Uncertainty Quantification". In: *Journal of Engineering Mathematics (Submitted)* (Oct. 2014), p. 26. arXiv: 1410.6338.

[9] D. Bigoni, A. P. Engsig-Karup, and Y. M. Marzouk. "Spectral tensor-train decomposition". In: *(Submitted)* (2014), p. 28. arXiv: 1405.5713.

[10] D. Bigoni. *Spectral Toolbox*. https://launchpad.net/spectraltoolbox. 2014.

[11] D. Bigoni. *Tensor Toolbox*. https://launchpad.net/tensortoolbox. 2014.

[12] D. Bigoni. *UQ Toolbox*. https://launchpad.net/uqtoolbox. 2014.

[13] Plato. *The Republic*. 380 B.C.

[14] Aristotele. *Metaphysics*. 384-322 B.C.

[15] A. Schopenhauer. *The World as Will and Representation*. 1818.

[16] E. Lorenz. "Deterministic nonperiodic flow". In: *Journal of the atmospheric sciences* (1963).

[17] B. Saltzman. "Finite amplitude free convection as an initial value problem-I". In: *Journal of the Atmospheric Sciences* (1962). DOI: http://dx.doi.org/10.1175/1520-0469(1962)019\%3C0329:FAFCAA\%3E2.0.CO;2.

[18] P. Billingsley. *Probability and measure*. 3rd. New York: John Wiley & Sons, 1995. ISBN: 0-471-00710-2.

[19] E. Hansen. *Measure theory*. 4th ed. Copenhagen: University of Copenhagen, 2009, p. 600. ISBN: 978-87-91927-44-7.

[20] G. Birkhoff and G. C. Rota. *Ordinary differential equations*. Wiley, 1978. ISBN: 9780471074113.

[21] W. A. Strauss. *Partial Differential Equations: An Introduction*. 2nd ed. John Wiley & Sons, Inc., 2008, p. 454. ISBN: 978-0470-05456-7.

[22] P. K. Kundu, I. M. Cohen, and D. R. Dowling. *Fluid Mechanics*. Academic Press. Academic Press, 2012. ISBN: 9780123821003.

[23] R. J. LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. en. Society for Industrial and Applied Mathematics, Sept. 2007, p. 341. ISBN: 9780898716290.

[24] R. Cook, D. S. Malkus, M. E. Plesha, and Robert J. Witt. *Concepts and applications of finite element analysis*. 4th ed. John Wiley & Sons, Inc., 2007.

[25] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Vol. 15. Texts in Applied Mathematics. New York, NY: Springer New York, 2008. ISBN: 978-0-387-75933-3. DOI: 10.1007/978-0-387-75934-0.

[26] C. Canuto, M. Hussaini, A. Quarteroni, and T. Zang. *Spectral methods - Fundamentals in Single Domains*. Springer-Verlag Berlin Heidelberg, 2006. ISBN: 9783540307259.

[27] C. Canuto, M. Hussaini, A. Quarteroni, and T. Zang. *Spectral methods: evolution to complex geometries and applications to fluid dynamics*. 2007.

[28] J. S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods*. Vol. 54. Texts in Applied Mathematics. New York, NY: Springer New York, 2008. ISBN: 978-0-387-72065-4. DOI: 10.1007/978-0-387-72067-8.

[29] D. A. Kopriva. *Implementing Spectral Methods for Partial Differential Equations*. Scientific Computation. Dordrecht: Springer Netherlands, 2009. ISBN: 978-90-481-2260-8. DOI: 10.1007/978-90-481-2261-5.

[30] G. Karniadakis and S. Sherwin. *Spectral/hp element methods for CFD*. Oxford: Oxford university press, 1999. ISBN: 0195102266.

[31] C. M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, 2006. ISBN: 9780387310732.

[32] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Vol. 1. Springer Series in Statistics, 2001. ISBN: 978-0-387-95284-0.

[33] W. Gautschi. *Orthogonal Polynomials: Computation and Approximation*. Numerical Mathematics and Scientific Computation. Oxford University Press, 2004. ISBN: 9781423771081.

[34] L. N. Trefethen. *Approximation Theory and Approximation Practice*. Siam, 2013. ISBN: 9781611972405.

[35] R. G. Ghanem and P. D. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Courier Dover Publications, Aug. 2003. ISBN: 9780486428185.

[36] O. L. P. L. Maître and O. M. Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*. en. Springer, June 2010, p. 542. ISBN: 9789048135196.

[37] D. Xiu. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, July 2010. ISBN: 9780691142128.

[38] OpenTURNS. *OpenTURNS - Reference Guide 1.0*. Tech. rep. 2013.

[39] A. D. Kiureghian and O. Ditlevsen. "Aleatory or epistemic? Does it matter?" In: *Structural Safety* 31.2 (Mar. 2009), pp. 105–112. ISSN: 01674730. DOI: 10.1016/j.strusafe.2008.06.020.

[40] N. Metropolis and S. Ulam. "The monte carlo method". In: *Journal of the American statistical association* 44.247 (1949), pp. 335–341.

[41] N. Wiener. "The homogeneous chaos". In: *American Journal of Mathematics* 60.4 (1938), pp. 897–936.

[42] H. Niederreiter. "Quasi-Monte Carlo methods and pseudo-random numbers". In: *Bulletin of the American Mathematical Society* 84.6 (1978), pp. 957–1041.

[43] S. Smolyak. "Quadrature and interpolation formulas for tensor products of certain classes of functions". In: *Dokl. Akad. Nauk SSSR* (1963).

[44] M. Mckay, R. Beckman, and W. Conover. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code". In: *Technometrics* 41.1 (2000), pp. 55–61.

[45] V. Eglajs and P. Audze. "New approach to the design of multifactor experiments". In: *Problems of Dynamics and Strength (Russian)* (1977).

[46] D. Xiu and G. E. Karniadakis. "The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations". In: *SIAM Journal on Scientific Computing* 24.2 (Jan. 2002), pp. 619–644. ISSN: 1064-8275. DOI: 10.1137/S1064827501387826.

[47] K. Petras. "Smolyak cubature of given polynomial degree with few nodes for increasing dimension". In: *Numerische Mathematik* 93.4 (Feb. 2003), pp. 729–753. ISSN: 0029-599X, 0945-3245.

[48] T. Gerstner and M. Griebel. "Dimension-adaptive tensor-product quadrature". In: *Computing* 71.1 (Aug. 2003), pp. 65–87. ISSN: 0010-485X. DOI: 10.1007/s00607-003-0015-5.

[49] F. Nobile, R. Tempone, and C. G. Webster. "An Anisotropic Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data". In: *SIAM Journal on Numerical Analysis* 46.5 (Jan. 2008), pp. 2411–2442. ISSN: 0036-1429. DOI: 10.1137/070680540. URL: http://epubs.siam.org/doi/abs/10.1137/070680540.

[50] P. G. Constantine, M. S. Eldred, and E. T. Phipps. "Sparse pseudospectral approximation method". In: *Computer Methods in Applied Mechanics and Engineering* 229-232 (July 2012), pp. 1–12. ISSN: 00457825. DOI: 10.1016/j.cma.2012.03.019.

[51] P. R. Conrad and Y. M. Marzouk. "Adaptive Smolyak Pseudospectral Approximations". In: *SIAM Journal on Scientific Computing* 35.6 (Jan. 2013), A2643–A2670. ISSN: 1064-8275. DOI: 10.1137/120890715. arXiv: arXiv:1209.1406v2. URL: http://arxiv.org/abs/1209.1406%20http://epubs.siam.org/doi/abs/10.1137/120890715.

[52] B. Øksendal. *Stochastic Differential Equations*. Vol. 29. Universitext 1. Berlin, Heidelberg: Springer Berlin Heidelberg, Mar. 2003, pp. 163–164. ISBN: 978-3-540-04758-2. DOI: 10.1007/978-3-642-14394-6.

[53] D. Revuz and M. Yor. *Continuous Martingales and Brownian Motion*. Vol. 75. 474. Springer Berlin Heidelberg, Dec. 1999. ISBN: 3-540-57622-3.

[54] G. De Marco. *Analisi Uno e Due*. Ed. by Decibel. 2nd ed. Padova: Zanichelli, 1999. ISBN: 978-88-08-01215-9.

[55] E. Kreyszig. *Introductory functional analysis with applications*. Wiley classics library. John Wiley & Sons, 2007. ISBN: 9788126511914.

[56] A. Kolmogorov and S. Fomin. *Introductory Real Analysis*. Ed. by R. A. Silverman. Revised En. New York: Dover Publication, Inc, 1970, p. 403. ISBN: 978-0-486-61226-3.

[57] J. Dongarra and A. L. Lastovetsky. *High Performance Heterogeneous Computing*. Wiley Series on Parallel and Distributed Computing. Wiley, 2009. ISBN: 9780470040393. URL: http://books.google.dk/books?id=puNfNQEACAAJ.

[58] R. Couturier. *Designing Scientific Applications on GPUs*. CRC Press / Taylor & Francis Group, 2014. ISBN: 9781466571624.

[59] M. Loève. *Probability Theory, vols. I-II*. 4th ed. Comprehensive Manuals of Surgical Specialties. New York: Springer, 1978. ISBN: 9780387902104.

[60] C. Schwab and R. A. Todor. "Karhunen-Loève approximation of random fields by generalized fast multipole methods". In: *Journal of Computational Physics* 217.1 (Sept. 2006), pp. 100–122. ISSN: 00219991. DOI: 10.1016/j.jcp.2006.01.048.

[61] G. Perrin, C. Soize, D. Duhamel, and C. Funfschilling. "Karhunen-Loève expansion revisited for vector-valued random fields: Scaling, errors and optimal basis." In: *Journal of Computational Physics* 242 (June 2013), pp. 607–622. ISSN: 00219991. DOI: 10.1016/j.jcp.2013.02.036.

[62] M. Rosenblatt. "Remarks on a Multivariate Transformation". In: *The Annals of Mathematical Statistics* 23.3 (1952), pp. 470–472. ISSN: 00034851.

[63] C. Desceliers, R. Ghanem, and C. Soize. "Maximum likelihood estimation of stochastic chaos representations from experimental data". In: *International Journal for Numerical Methods in Engineering* 66.6 (May 2006), pp. 978–1001. ISSN: 0029-5981. DOI: 10.1002/nme.1576.

[64] M. Arnst, R. Ghanem, and C. Soize. "Identification of Bayesian posteriors for coefficients of chaos expansions". In: *Journal of Computational Physics* 229.9 (May 2010), pp. 3134–3154. ISSN: 00219991. DOI: 10.1016/j.jcp.2009.12.033.

[65] C. Soize. "Identification of high-dimension polynomial chaos expansions with random coefficients for non-Gaussian tensor-valued random fields using partial and limited experimental data". In: *Computer Methods in Applied Mechanics and Engineering* 199.33-36 (July 2010), pp. 2150–2164. ISSN: 00457825. DOI: 10.1016/j.cma.2010.03.013.

[66] G. Perrin, C. Soize, D. Duhamel, and C. Funfschilling. "Identification of Polynomial Chaos Representations in High Dimension from a Set of Realizations". In: *SIAM Journal on Scientific Computing* 34.6 (Jan. 2012), A2917–A2945. ISSN: 1064-8275. DOI: 10.1137/11084950X. URL: http://epubs.siam.org/doi/abs/10.1137/11084950X.

[67] C. Soize. "Construction of probability distributions in high dimension using the maximum entropy principle: Applications to stochastic processes, random fields and random". In: *International Journal for Numerical Methods in Engineering* July (2008), pp. 1583–1611. DOI: 10.1002/nme.

[68] N.-B. Heidenreich, A. Schindler, and S. Sperlich. "Bandwidth selection for kernel density estimation: a review of fully automatic selectors". In: *AStA Advances in Statistical Analysis* 97.4 (June 2013), pp. 403–433. ISSN: 1863-8171. DOI: 10.1007/s10182-013-0216-y.

[69] J. D. Cawlfield. "Reliability Algorithms: FORM and SORM Methods". In: *Sensitivity Analysis*. Ed. by A. Saltelli, K. Chan, and E. M. Scott. Chichester: John Wiley & Sons, Ltd., 2000.

[70] J. D. Cawlfield, S. Boateng, J. Piggott, and M.-C. Wu. "Probabilistic sensitivity measures applied to numerical models of flow and transport". In: *Journal of Statistical Computation and Simulation* 57.1-4 (Apr. 1997), pp. 353–364. ISSN: 0094-9655. DOI: 10.1080/00949659708811817.

[71] R. E. Melchers. *Structural Reliability: Analysis and Prediction*. Ellis Horwood series in civil engineering. Ellis Horwood, John Wiley, 1987. ISBN: 9780853129301.

[72] J. E. Gentle. *Random number generation and Monte Carlo methods*. Springer, 2003, 380 s. ISBN: 0387001786, 9780387001784.

[73] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. 3rd ed. New York, NY, USA: Cambridge University Press, 2007. ISBN: 0521880688, 9780521880688.

[74] B. Tang. "Orthogonal Array-Based Latin Hypercubes". In: *Journal of the American Statistical Association* 88.424 (Dec. 1993), p. 1392. ISSN: 01621459. DOI: 10.2307/2291282.

[75] B. A. Finlayson. *The Method of Weighted Residuals and Variational Principles - With Application in Fluid Mechanics, Heat and Mass Transfer*. Vol. 87. Mathematics in Science and Engineering. Elsevier, 1972. ISBN: 9780122570506.

[76] A. Narayan and D. Xiu. "Stochastic Collocation Methods on Unstructured Grids in High Dimensions via Interpolation". In: *SIAM Journal on Scientific Computing* 34.3 (Jan. 2012), A1729–A1752. ISSN: 1064-8275. DOI: 10.1137/110854059.

[77] C. De Boor and A. Ron. "Computational Aspects of Polynomial Interpolation in Several Variables". In: *Mathematics of Computation* 58.198 (Apr. 1992), p. 705. ISSN: 00255718. DOI: 10.2307/2153210.

[78] C. De Boor and A. Ron. "On multivariate polynomial interpolation". In: *Constructive Approximation* 6.3 (Sept. 1990), pp. 287–302. ISSN: 0176-4276. DOI: 10.1007/BF01890412.

[79] N. Cressie. "The origins of kriging". In: *Mathematical Geology* 22.3 (Apr. 1990), pp. 239–252. ISSN: 0882-8121. DOI: 10.1007/BF00889887.

[80] R. H. Cameron and W. T. Martin. "The Orthogonal Development of Non-Linear Functionals in Series of Fourier-Hermite Functionals". In: *The Annals of Mathematics* 48.2 (Apr. 1947), p. 385. ISSN: 0003486X. DOI: 10.2307/1969178.

[81] R. Koekoek, P. A. Lesky, and R. F. Swarttouw. *Hypergeometric Orthogonal Polynomials and Their q-Analogues*. Springer Monographs in Mathematics. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. ISBN: 978-3-642-05013-8. DOI: 10.1007/978-3-642-05014-5.

[82] W. Gautschi. "Algorithm 726: ORTHPOL-a package of routines for generating orthogonal polynomials and Gauss-type quadrature rules". In: *ACM Transactions on Mathematical Software (TOMS)* 20.1 (Mar. 1994), pp. 21–62. ISSN: 0098-3500. DOI: 10.1145/174603.174605.

[83] M. Gerritsma, J. van der Steen, P. Vos, and G. Karniadakis. "Time-dependent generalized polynomial chaos". In: *Journal of Computational Physics* 229.22 (Nov. 2010), pp. 8333–8363. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2010.07.020.

[84] O. Le Maître, O. Knio, B. Debusschere, H. Najm, and R. Ghanem. "A multigrid solver for two-dimensional stochastic diffusion equations". In: *Computer Methods in Applied Mechanics and Engineering* 192.41-42 (Oct. 2003), pp. 4723–4744. ISSN: 00457825. DOI: 10.1016/S0045-7825(03)00457-2.

[85] M. Pellissetti and R. Ghanem. "Iterative solution of systems of linear equations arising in the context of stochastic finite elements". In: *Advances in Engineering Software* 31.8-9 (Aug. 2000), pp. 607–616. ISSN: 09659978. DOI: 10.1016/S0965-9978(00)00034-X.

[86] G. Poëtte and D. Lucor. "Non intrusive iterative stochastic spectral representation with application to compressible gas dynamics". In: *Journal of Computational Physics* 231 (2012), pp. 3587–3609. DOI: 10.1016/j.jcp.2011.12.038.

[87] J. Foo, X. Wan, and G. E. Karniadakis. "The multi-element probabilistic collocation method (ME-PCM): Error analysis and applications". In: *Journal of Computational Physics* 227.22 (2008), pp. 9572–9595.

[88] X. Wan and G. E. Karniadakis. "Multi-Element Generalized Polynomial Chaos for Arbitrary Probability Measures". In: *SIAM Journal on Scientific Computing* 28.3 (2006), p. 901. ISSN: 10648275. DOI: 10.1137/050627630.

[89] X. Wan and G. E. Karniadakis. "An adaptive multi-element generalized polynomial chaos method for stochastic differential equations". In: *Journal of Computational Physics* 209.2 (2005), pp. 617–642.

[90] O. Le Maître, O. Knio, H. Najm, and R. Ghanem. "Uncertainty propagation using Wiener-Haar expansions". In: *Journal of Computational Physics* 197.1 (June 2004), pp. 28–57. ISSN: 00219991. DOI: 10.1016/j.jcp.2003.11.033.

[91] O. Le Maître, H. Najm, R. Ghanem, and O. Knio. "Multi-resolution analysis of Wiener-type uncertainty propagation schemes". In: *Journal of Computational Physics* 197.2 (July 2004), pp. 502–531. ISSN: 00219991. DOI: 10.1016/j.jcp.2003.12.020.

[92] O. P. Le Maître, H. N. Najm, P. P. Pébay, R. G. Ghanem, and O. M. Knio. "Multi-Resolution-Analysis Scheme for Uncertainty Quantification in Chemical Systems". In: *SIAM Journal on Scientific Computing* 29.2 (Jan. 2007), pp. 864–889. ISSN: 1064-8275. DOI: 10.1137/050643118.

[93] E. Hairer, G. Wanner, and S. P. Nørsett. *Solving Ordinary Differential Equations I: Nonstiff problems*. Vol. 8. Springer Series in Computational Mathematics. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993. ISBN: 978-3-540-56670-0. DOI: 10.1007/978-3-540-78862-1.

[94] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 1996. ISBN: 9780801854149.

[95] R. Bellman and R. Corporation. *Dynamic Programming*. Rand Corporation research study. Princeton University Press, 1957. ISBN: 9780691079516.

[96] A. H. Stroud. "Remarks on the Disposition of Points in Numerical Integration Formulas". In: *Mathematical Tables and Other Aids to Computation* 11.60 (Oct. 1957), pp. 257–261. ISSN: 0891-6837. DOI: 10.2307/2001945.

[97] A. H. Stroud. "Numerical Integration Formulas of Degree Two". In: *Mathematics of Computation* 14.69 (Jan. 1960), p. 21. ISSN: 00255718. DOI: 10.2307/2002981.

[98] D. Xiu. "Numerical integration formulas of degree two". In: *Applied Numerical Mathematics* 58.10 (Oct. 2008), pp. 1515–1520. ISSN: 01689274. DOI: 10.1016/j.apnum.2007.09.004.

[99] A. S. Kronrod. "Nodes and Weights of Quadrature Formulas". In: *English transl. from Russian, Consultants Bureau* 35.597 (1965).

[100] T. N. L. Patterson. "The Optimum Addition of Points to Quadrature Formulae". In: *Mathematics of Computation* 22.104 (Oct. 1968), 847–s31. ISSN: 0025-5718. DOI: 10.2307/2004583.

[101] C. W. Clenshaw and A. R. Curtis. "A method for numerical integration on an automatic computer". In: *Numerische Mathematik* 2.1 (1960), pp. 197–205. ISSN: 0029599X. DOI: 10.1007/BF01386223.

[102] J. Waldvogel. "Fast Construction of the Fejér and Clenshaw-Curtis Quadrature Rules". In: *Bit Numerical Mathematics* 46.1 (2006), pp. 195–202. ISSN: 00063835. DOI: 10.1007/s10543-006-0045-4.

[103] L. Fejér. "Mechanische Quadraturen mit positiven Cotesschen Zahlen". In: *Math. Z.* 37 (1933), pp. 287–309.

[104] T. Gerstner and M. Griebel. "Numerical integration using sparse grids". In: *Numerical algorithms* 18 (1998), pp. 209–232.

[105] M. Fornasier and R. Holger. "Compressive sensing". In: *Signal Processing Magazine, IEEE* (2007), pp. 1–49.

[106] H. Rauhut and R. Ward. "Sparse Legendre expansions via ℓ_1-minimization". In: *Journal of Approximation Theory* 164.5 (May 2012), pp. 517–533. ISSN: 00219045. DOI: 10.1016/j.jat.2012.01.008.

[107] L. Yan, L. Guo, and D. Xiu. "Stochastic collocation algorithms using ℓ_1-minimization". In: *International Journal for Uncertainty Quantification* 2.3 (2012), pp. 279–293. ISSN: 2152-5080. DOI: 10.1615/Int.J.UncertaintyQuantification.2012003925.

[108] X. Yang and G. E. Karniadakis. "Reweighted ℓ_1 minimization method for stochastic elliptic differential equations". In: *Journal of Computational Physics* 248 (Sept. 2013), pp. 87–108. ISSN: 00219991. DOI: 10.1016/j.jcp.2013.04.004.

[109] R. Li and R. Ghanem. "Adaptive polynomial chaos expansions applied to statistics of extremes in nonlinear random vibration". In: *Probabilistic Engineering Mechanics* 13.2 (Apr. 1998), pp. 125–136. ISSN: 02668920. DOI: 10.1016/S0266-8920(97)00020-9.

[110] J. Le Meitour, D. Lucor, and J.-C. Chassaing. "Prediction of stochastic limit cycle oscillations using an adaptive Polynomial Chaos method". In: *Journal of Aeroelasticity and Structural Dynamics* 2.1 (2010), pp. 3–22. DOI: 10.3293/asdj.2010.4.

[111] D. Lucor and G. E. Karniadakis. "Adaptive Generalized Polynomial Chaos for Nonlinear Random Oscillators". In: *SIAM Journal on Scientific Computing* 26.2 (Jan. 2004), pp. 720–735. ISSN: 1064-8275. DOI: 10.1137/S1064827503427984.

[112] N. Aubry. "On the hidden beauty of the proper orthogonal decomposition". In: *Theoretical and Computational Fluid Dynamics* 900265 (1991), pp. 339–352.

[113] N. Aubry, R. Guyonnet, and R. Lima. "Spatiotemporal analysis of complex signals: Theory and applications". In: *Journal of Statistical Physics* 64.3-4 (Aug. 1991), pp. 683–739. ISSN: 0022-4715. DOI: 10.1007/BF01048312.

[114] F. Chinesta, R. Keunings, and A. Leygue. *The Proper Generalized Decomposition for Advanced Numerical Simulations*. SpringerBriefs in Applied Sciences and Technology. Cham: Springer International Publishing, 2014. ISBN: 978-3-319-02864-4. DOI: 10.1007/978-3-319-02865-1.

[115] A. Nouy. "A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations". In: *Computer Methods in Applied Mechanics and Engineering* 196.45-48 (Sept. 2007), pp. 4521–4537. ISSN: 00457825. DOI: 10.1016/j.cma.2007.05.016.

[116] A. Nouy. "Proper Generalized Decompositions and Separated Representations for the Numerical Solution of High Dimensional Stochastic Problems". In: *Archives of Computational Methods in Engineering* 17.4 (2010), pp. 403–434. ISSN: 1134-3060. DOI: 10.1007/s11831-010-9054-1.

[117] L. Tamellini, O. Le Maître, and A. Nouy. "Model Reduction Based on Proper Generalized Decomposition for the Stochastic Steady Incompressible Navier–Stokes Equations". In: *SIAM Journal on Scientific Computing* 36.3 (Jan. 2014), A1089–A1117. ISSN: 1064-8275. DOI: 10.1137/120878999.

[118] T. P. Sapsis and P. F. Lermusiaux. "Dynamically orthogonal field equations for continuous stochastic dynamical systems". In: *Physica D: Nonlinear Phenomena* 238.23-24 (Dec. 2009), pp. 2347–2360. ISSN: 01672789. DOI: 10.1016/j.physd.2009.09.017.

[119] M. P. Ueckermann, T. P. Sapsis, and P. F. J. Lermusiaux. "Numerical Schemes for Dynamically Orthogonal Equations of Stochastic Fluid and Ocean Flows". In: *Journal of Computational Physics* (2011).

[120] B. Khoromskij and I. Oseledets. "Quantics-TT collocation approximation of parameter-dependent and stochastic elliptic PDEs". In: *Comput. Methods Appl. Math.* (2010).

[121] B. Khoromskij and C. Schwab. "Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs". In: *SIAM Journal on Scientific Computing* 33.1 (2011), pp. 364–385.

[122] I. Oseledets. "Tensor-train decomposition". In: *SIAM Journal on Scientific Computing* 33.5 (2011), pp. 2295–2317.

[123] T. Kolda and B. Bader. "Tensor decompositions and applications". In: *SIAM review* 51.3 (Aug. 2009), pp. 455–500. ISSN: 0036-1445, 1095-7200. DOI: 10.1137/07070111X.

[124] L. Grasedyck, D. Kressner, and C. Tobler. "A literature survey of low-rank tensor approximation techniques". In: *arXiv preprint arXiv:1302.7121* (2013), pp. 1–20. arXiv: arXiv:1302.7121v1.

[125] L. Grasedyck. "Hierarchical singular value decomposition of tensors". In: *SIAM Journal on Matrix Analysis and Applications* 31.4 (2010), pp. 2029–2054.

[126] I. Oseledets and E. Tyrtyshnikov. "Tensor tree decomposition does not need a tree". In: *Submitted to Linear Algebra Appl* (2009).

[127] P. R. Halmos and V. S. Sunder. *Bounded integral operators on L2 spaces*. Ergebnisse der Mathematik und ihrer Grenzgebiete. Springer-Verlag, 1978. ISBN: 9783540088943.

[128] L. N. Trefethen and D. Bau III. *Numerical linear algebra*. Vol. 12. Society for Industrial and Applied Mathematics, 1997, p. 361. ISBN: 0898713617.

[129] A. Townsend and L. N. Trefethen. "Continuous analogues of matrix factorizations". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 471.2173 (Nov. 2014), pp. 20140585–20140585. ISSN: 1364-5021. DOI: 10.1098/rspa.2014.0585.

[130] F. Smithies. "The eigen-values and singular values of integral equations". In: *Proceedings of the London Mathematical Society*. 5. 1937.

[131] G. H. Hardy and J. E. Littlewood. "Some new properties of fourier constants". In: *Mathematische Annalen* 97.1 (Dec. 1927), pp. 159–209. ISSN: 0025-5831. DOI: 10.1007/BF01447865.

[132] K. Jörgens. *Linear integral operators*. Surveys and reference works in mathematics. Pitman Advanced Pub. Program, 1982. ISBN: 9780273085232.

[133] A. Hammerstein. "Über die Entwickelung des Kernes linearer Integralgleichu ngen nach Eigenfunktionen". In: *Sitzungsberichte Preuss. Akad. Wiss.* (1923), pp. 181–184.

[134] D. Savostyanov and I. Oseledets. "Fast adaptive interpolation of multi-dimensional arrays in tensor train format". In: *The 2011 International Workshop on Multidimensional (nD) Systems* (Sept. 2011), pp. 1–8. DOI: 10.1109/nDS.2011.6076873.

[135] D. Savostyanov. "Quasioptimality of maximum-volume cross interpolation of tensors". In: *arXiv preprint arXiv:1305.1818* c (2013), pp. 1–23. arXiv: arXiv:1305.1818v2.

[136] B. Khoromskij. "O (dlog N)-Quantics Approximation of Nd Tensors in High-Dimensional Numerical Modeling". In: *Constructive Approximation* (2011), pp. 257–280. DOI: 10.1007/s00365-011-9131-1.

[137] B. Khoromskij and I. Oseledets. "QTT approximation of elliptic solution operators in higher dimensions". In: *Russian Journal of Numerical Analysis and Mathematical Modelling* 26.3 (2011), pp. 303–322.

[138] A. Genz. "A package for testing multiple integration subroutines". In: *Numerical Integration* (1987). Ed. by P. Keast and G. Fairweather. DOI: 10.1007/978-94-009-3889-2.

[139] A. Genz. "Testing multidimensional integration routines". In: *Proc. of international conference on Tools, methods and languages for scientific and engineering computation* (1984), pp. 81–94.

[140] Z. Zhang, X. Yang, I. V. Oseledets, G. E. Karniadakis, and L. Daniel. "Enabling High-Dimensional Hierarchical Uncertainty Quantification by ANOVA and Tensor-Train Decomposition". In: (July 2014), p. 13. arXiv: 1407.3023.

[141] D. L. Applegate, R. E. Bixby, V. Chv\'{a}tal, and W. J. Cook. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press, 2007, p. 608. ISBN: 9781400841103.

[142] H. Rabitz and O. Alis. "Managing the Tyranny of Parameters in Mathematical Modelling of Physical Systems". In: *Sensitivity Analysis*. Ed. by A. Saltelli, K. Chan, and E. M. Scott. Chichester: John Wiley & Sons, Ltd., 2000.

[143] K. Chan, S. Tarantola, A. Saltelli, and I. Sobol'. "Variance-based methods". In: *Sensitivity Analysis*. Ed. by A. Saltelli, K. Chan, and E. M. Scott. Chichester: John Wiley & Sons, Ltd., 2000.

[144] I. Sobol'. "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates". In: *Mathematics and Computers in Simulation* 55.1-3 (Feb. 2001), pp. 271–280. ISSN: 03784754. DOI: 10.1016/S0378-4754(00)00270-6.

[145] Z. Zhang, M. Choi, and G. E. Karniadakis. "Anchor Points Matter in ANOVA Decomposition". In: *Selected papers from the ICOSAHOM '09 conference, June 22-26, Trondheim, Norway*. Ed. by J. S. Hesthaven and E. M. Rønquist. Vol. 76. Lecture Notes in Computational Science and Engineering. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 347–355. ISBN: 978-3-642-15336-5. DOI: 10.1007/978-3-642-15337-2.

[146] Z. Gao and J. Hesthaven. "Efficient solution of ordinary differential equations with high-dimensional parametrized uncertainty". In: *Communications in Computational Physics* 10.2 (2011), pp. 253–286.

[147] T. Crestaux, O. Le Maître, and J.-M. Martinez. "Polynomial chaos expansion for sensitivity analysis". In: *Reliability Engineering & System Safety* 94.7 (July 2009), pp. 1161–1172. ISSN: 09518320. DOI: 10.1016/j.ress.2008.10.008.

[148] R. M. Errico. "What Is an Adjoint Model?" In: *Bulletin of the American Meteorological Society* (1997), pp. 2577–2591.

[149] M. Ulbrich and S. Ulbrich. *Primal-dual interior-point methods for PDE-constrained optimization*. Vol. 117. 1-2. July 2007, pp. 435–485. ISBN: 1010700701687. DOI: 10.1007/s10107-007-0168-7.

[150] Y. Cao, S. Li, L. Petzold, and R. Serban. "Adjoint Sensitivity Analysis for Differential-Algebraic Equations: The Adjoint DAE System and Its Numerical Solution". In: *SIAM Journal on Scientific Computing* 24.3 (Jan. 2003), pp. 1076–1089. ISSN: 1064-8275. DOI: 10.1137/S1064827501380630.

[151] R. Herzog and K. Kunisch. "Algorithms for PDE-constrained optimization". In: *GAMM-Mitteilungen* 33.2 (Oct. 2010), pp. 163–176. ISSN: 09367195. DOI: 10.1002/gamm.201010013.

[152] T. Maly and L. R. Petzold. "Numerical methods and software for sensitivity analysis of differential-algebraic systems". In: *Applied Numerical Mathematics* 20.60 (1996), pp. 57–79.

[153] C. Bischof, A. Carle, G. Corliss, A. Griewank, and P. Hovland. "ADIFOR – Generating Derivative Codes from Fortran Programs". In: *Scientific Programming* 1.1 (1992), pp. 1–22.

[154] I. Sobol'. "Sensitivity analysis for non linear mathematical models". In: *Math. Model. Comput. Exp.* 1 (1993), pp. 407–414.

[155] A. Saltelli, S. Tarantola, and K. P.-S. Chan. "A Quantitative Model-Independent Method for Global Sensitivity Analysis of Model Output". In: *Technometrics* 41.1 (Feb. 1999), pp. 39–56. ISSN: 0040-1706. DOI: 10.1080/00401706.1999.10485594.

[156] C. P. Robert. *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. Ed. by I. O. George Casella Stephen Fienberg. Vol. 91. Springer texts in statistics 433. Springer, 2007, p. 602. ISBN: 9780387715988. DOI: 10.1007/0-387-71599-1.

[157] A. Tarantola. *Inverse problem theory and methods for model parameter estimation*. English. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2005. ISBN: 0898715725 9780898715729.

[158] S. Brooks, A. Gelman, G. Jones, and X. Meng. *Handbook of Markov Chain Monte Carlo*. Ed. by S. Brooks, A. Gelman, G. Jones, and X. Meng. Chapman and Hall/CRC, 2011. ISBN: 9781420079418.

[159] C. Geyer. "Introduction to Markov Chain Monte Carlo". In: *Handbook of Markov Chain Monte Carlo*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. Chapman and Hall/CRC, May 2011. ISBN: 978-1-4200-7941-8. DOI: doi:10.1201/b10905-2.

[160] S. Evans and P. Stark. "Inverse problems as statistics". In: *Inverse problems* 55 (2002).

[161] Y. M. Marzouk and H. N. Najm. "Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems". In: *Journal of Computational Physics* 228.6 (Apr. 2009), pp. 1862–1902. ISSN: 00219991. DOI: 10.1016/j.jcp.2008.11.024.

[162] Y. M. Marzouk, H. N. Najm, and L. A. Rahn. "Stochastic spectral methods for efficient Bayesian solution of inverse problems". In: *Journal of Computational Physics* 224.2 (June 2007), pp. 560–586. ISSN: 00219991. DOI: 10.1016/j.jcp.2006.10.010.

[163] EN14363. *Railway applications - Testing for the acceptance of running characteristics of railway vehicles - Testing of running behaviour and stationary tests*. Tech. rep. Brussels, 2005, p. 113.

[164] C. Funfschilling, G. Perrin, and S. Kraft. "Propagation of variability in railway dynamic simulations: application to virtual homologation". In: *Vehicle System Dynamics* 50.sup1 (2012), pp. 245–261. ISSN: 0042-3114. DOI: 10.1080/00423114.2012.676757.

[165] C. Funfschilling, Y. Bezin, and M. Sebès. "DynoTRAIN: introduction of simulation in the certification process of railway vehicles". In: *Transport Research Arena*. Paris, 2014.

[166] G. Perrin, C. Soize, D. Duhamel, and C. Funfschilling. "Track irregularities stochastic modeling". In: *Probabilistic Engineering Mechanics* 34 (Oct. 2013), pp. 123–130. ISSN: 02668920. DOI: 10.1016/j.probengmech.2013.08.006.

[167] G. Perrin, C. Soize, D. Duhamel, and C. Funfschilling. "Dynamical behavior of trains excited by a non-Gaussian vector-valued random field". In: *COMPDYN 2013, 4th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquacke Engineering*. Ed. by M. Papadrakakis, V. Papadopoulos, and V. Plevris. Kos Island, Greece, 2013.

[168] N. Cooperrider. "The Hunting Behavior of conventional Railway Trucks". In: *ASME J. Engineering and Industry* 94 (1972), pp. 752–762.

[169] H. True and C. Kaas-Petersen. "A Bifurcation Analysis of Nonlinear Oscillations in Railway Vehicles". In: *Vehicle System Dynamics* 12 (July 1983), pp. 5–6. ISSN: 0042-3114. DOI: 10.1080/00423118308965288.

[170] D. Bigoni. *Curving Dynamics in High Speed Trains*. Master Thesis Supervised by Associate Professor Allan Peter Engsig-Karup, apek@imm.dtu.dk, DTU Informatics. Asmussens Alle, Building 305, DK-2800 Kgs. Lyngby, Denmark, 2011.

[171] W. Kik and D. Moelle. *ACRadSchiene - To create or Approximate Wheel/Rail profiles - Tutorial*. Tech. rep. 2007.

[172] J. Kalker. "Wheel-rail rolling contact theory". In: *Wear* 144.1-2 (Apr. 1991), pp. 243–261. ISSN: 00431648. DOI: 10.1016/0043-1648(91)90018-P.

[173] Z. Y. Shen, J. K. Hedrick, and J. A. Elkins. "A Comparison of Alternative Creep Force Models for Rail Vehicle Dynamic Analysis". In: *Vehicle System Dynamics* 12.1-3 (July 1983), pp. 79–83. ISSN: 0042-3114. DOI: 10.1080/00423118308968725.

[174] S. H. Strogatz. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Advanced book program. Westview Press, 1994. ISBN: 9780738204536. URL: http://books.google.dk/books?id=FIYHiBLWCJMC.

[175] H. True. "On the theory of nonlinear dynamics and its applications in vehicle systems dynamics". In: *Vehicle System Dynamics* 31.5-6 (1999), pp. 393–421. ISSN: 0042-3114. DOI: 10.1076/vesd.31.5.393.8361.

[176] F. Xia and H. True. "On the dynamics of the three-piece-freight truck". In: *Proceedings of the 2003 IEEE/ASME Joint Railroad Conference, 2003.* IEEE, 2003, pp. 149–159. ISBN: 0-7803-7741-9. DOI: 10.1109/RRCON.2003.1204661.

[177] H. True and L. Trzepacz. "On the Dynamics of a Railway Freight Wagon Wheelset with Dry Friction Damping". In: *IUTAM Symposium on Chaotic Dynamics and Control of Systems and Processes in Mechanics*. Ed. by G. Rega and F. Vestroni. Vol. 122. Berlin/Heidelberg: Springer-Verlag, 2005, pp. 159–168. ISBN: 1-4020-3267-6.

[178] H. True. "Multiple attractors and critical parameters and how to find them numerically: the right, the wrong and the gambling way". In: *Vehicle System Dynamics* 51.3 (2013), pp. 443–459. ISSN: 0042-3114. DOI: 10.1080/00423114.2012.738919.

[179] L. Mazzola and S. Bruni. "Effect of Suspension Parameter Uncertainty on the Dynamic Behaviour of Railway Vehicles". In: *Applied Mechanics and Materials* 104 (Sept. 2011), pp. 177–185. ISSN: 1662-7482. DOI: 10.4028/www.scientific.net/AMM.104.177.

[180] G. Kewlani, J. Crawford, and K. Iagnemma. "A polynomial chaos approach to the analysis of vehicle dynamics under uncertainty". In: *Vehicle System Dynamics* 50.5 (May 2012), pp. 749–774. ISSN: 0042-3114, 1744-5159. DOI: 10.1080/00423114.2011.639897.

[181] E. M. Bitner-Gregersen and Ø. Hagen. "Uncertainties in data for the offshore environment". In: *Structural Safety* 7.1 (Jan. 1990), pp. 11–34. ISSN: 01674730. DOI: 10.1016/0167-4730(90)90010-M. URL: http://linkinghub.elsevier.com/retrieve/pii/016747309090010M.

[182] E. M. Bitner-Gregersen, S. K. Bhattacharya, I. K. Chatjigeorgiou, I. Eames, K. Ellermann, K. Ewans, G. Hermanski, M. C. Johnson, N. Ma, C. Maisondieu, A. Nilva, I. Rychlik, and T. Waseda. "Recent developments of ocean environmental description with focus on uncertainties". In: *Ocean Engineering* 86 (Aug. 2014), pp. 26–46. ISSN: 00298018. DOI: 10.1016/j.oceaneng.2014.03.002.

[183] E. M. Bitner-Gregersen, K. C. Ewans, and M. C. Johnson. "Some uncertainties associated with wind and wave description and their importance for engineering applications". In: *Ocean Engineering* 86 (Aug. 2014), pp. 11–25. ISSN: 00298018. DOI: 10.1016/j.oceaneng.2014.05.002.

[184] A. P. Engsig-Karup, S. L. Glimberg, and A. S. Nielsen. "Fast hydrodynamics on heterogeneous many-core hardware". In: *Designing Scientific Applications on GPUs*. Chapman & Hall/CRC Numerical Analy & Scient Comp. Series. Chapman and Hall/CRC, Nov. 2013, pp. 251–294. ISBN: 978-1-4665-7162-4. DOI: 10.1201/b16051-17.

[185] L. Ge, K. F. Cheung, and M. H. Kobayashi. "Stochastic Solution for Uncertainty Propagation in Nonlinear Shallow-Water Equations". In: *Journal of Hydraulic Engineering* 134.12 (Dec. 2008), pp. 1732–1743. ISSN: 0733-9429. DOI: 10.1061/(ASCE)0733-9429(2008)134:12(1732).

[186] D. Liu. "Uncertainty Quantification with Shallow Water Equations". PhD thesis. University of Braunschweig – Institute of Technology, 2009.

[187] M. Ricchiuto, P. M. Congedo, and A. Delis. *Runup and uncertainty quantification: sensitivity analysis via ANOVA decomposition*. Tech. rep. April. Bordeaux: INRIA, 2014.

[188] A. Naess and T. Moan. *Stochastic Dynamics of Marine Structures*. Cambridge: Cambridge University Press, 2012. ISBN: 9781139021364. DOI: 10.1017/CBO9781139021364.

[189] W. He, M. Diez, Z. Zou, E. F. Campana, and F. Stern. "URANS study of Delft catamaran total/added resistance, motions and slamming loads in head sea including irregular wave and uncertainty quantification for variable regular wave and geometry". In: *Ocean Engineering* 74 (Dec. 2013), pp. 189–217. ISSN: 00298018. DOI: 10.1016/j.oceaneng.2013.06.020.

[190] P. Jonathan and K. Ewans. "Statistical modelling of extreme ocean environments for marine design: A review". In: *Ocean Engineering* 62 (Apr. 2013), pp. 91–109. ISSN: 00298018. DOI: 10.1016/j.oceaneng.2013.01.004.

[191] G. B. Whitham. *Linear and Nonlinear Waves*. A Wiley-Interscience publication. Wiley, 1974. ISBN: 9780471940906.

[192] H. B. Bingham and H. Zhang. "On the accuracy of finite-difference solutions for nonlinear water waves". In: *Journal of Engineering Mathematics* 58 (2007), pp. 211–228.

[193] A. Engsig-Karup, H. Bingham, and O. Lindberg. "An efficient flexible-order model for 3D nonlinear water waves". In: *Journal of Computational Physics* 228.6 (Apr. 2009), pp. 2100–2118. ISSN: 00219991. DOI: 10.1016/j.jcp.2008.11.028.

[194] A. Engsig-Karup, M. G. Madsen, and S. L. Glimberg. "A massively parallel GPU-accelerated model for analysis of fully nonlinear free surface waves". In: *International Journal for Numerical Methods in Fluids* 70.1 (2011), pp. 20–36. DOI: 10.1002/fld.

[195] S. L. Glimberg, A. P. Engsig-Karup, and M. G. Madsen. "A Fast GPU-accelerated Mixed-precision Strategy for Fully Nonlinear Water Wave Computations". In: *Numerical Mathematics and Advanced Applications 2011, Proceedings of ENUMATH 2011, the 9th European Conference on Numerical Mathematics and Advanced Applications, Leicester, September 2011*. Ed. by A. C. Et al. Springer, 2012.

[196] S. Beji and J. A. Battjes. "Numerical simulation of nonlinear-wave propagation over a bar". In: *Coastal Engineering* 23 (1994), pp. 1–16.

[197] H. R. Luth, B. Klopman, and N. Kitou. "Projects 13G: Kinematics of waves breaking partially on an offshore bar: LDV measurements for waves with and without a net onshore current". In: *{\em Technical report H1573}, Delft Hydraulics* (1994).

[198] L. Benxia and Y. Xiping. "Wave decomposition phenomenon and spectrum evolution over submerged bars". In: *Acta Oceanologica Sinica* 28.3 (2009), pp. 82–92.

[199] R. W. Whalin. *The limit of applicability of linear wave refraction theory in convergence zone*. Tech. rep. H-71-3. US Army Corps of Engineers, 1971.

[200] L. C. Evans. *Partial Differential Equations*. Graduate studies in mathematics. American Mathematical Society, 1998. ISBN: 9780821807729.

[201] W. Gautschi. "Construction of Gauss-Christoffel Quadrature Formulas". In: *Mathematics of Computation* 22.102 (Apr. 1968), p. 251. ISSN: 00255718. DOI: 10.2307/2004654.

[202] J. Shen and L. L. Wang. "Some recent advances on spectral methods for unbounded domains". In: *Communications in Computational Physics* 5.2-4 (2009), pp. 195–241.

[203] L. S. Glimberg, A. P. Engsig-Karup, A. S. Nielsen, and B. Dammann. "Development of software components for heterogeneous many-core architectures". In: *Designing Scientific Applications on GPUs*. Ed. by R. Couturier. Lecture notes in computational science and engineering. CRC Press / Taylor & Francis Group, 2013, pp. 73–104.

[204] I. Oseledets and E. Tyrtyshnikov. "TT-cross approximation for multidimensional arrays". In: *Linear Algebra and its Applications* 432.1 (Jan. 2010), pp. 70–88. ISSN: 00243795. DOI: 10.1016/j.laa.2009.07.024.

Index